SEMANTIC MODELS

Fred Landman

Spring 2013
Version of 2019

CONTENTS

1. UNIVERSAL ALGEBRA

Let A be a set.

pow(A) is the powerset of A, pow(A) = $\{X: X \subseteq A\}$

$$A^n = \underbrace{A \times \ldots \times A}_{n \text{ times}}$$

For n-tuple $<a_1,\ldots a_k,\ldots a_n>$ we define: $<a_1,\ldots a_k,\ldots a_n>^k = a_k$

An n-place **relation** on A is a subset of pow(A^n)

Let R be an n-place relation:

The **k-th projection of** R, $\pi_k(R)$ is:

$$\pi_k(R) = \{x_k:\ \exists x_1,\ldots\exists x_{k-1}\exists x_{k+1}\ldots\exists x_n: <x_1,\ldots,x_{k-1}x_kx_{k+1}\ldots x_n> \in R\}$$

If R is a two=place relation, $\pi_1(R)$= **dom**(R) (domain) and $\pi_2(R)$=**ran**(R) (range).

If R is an n-place relation, the **converse relation**, R^{-1}, is: $\{<y,x>: <x,y> \in R\}$

Let A and B be sets.

A **function from** A **into** B is a relation f such that:

1. dom(f) = A
2. ran(f) $\subseteq$ B
3. For every $a \in A$, $b_1,b_2 \in B$ if $<a,b_1>,<a,b_2> \in f$ then $b_1 = b_2$

We write f: A $\rightarrow$ B for function f from A into B and f(a)=b for $<a,b> \in f$.

We have defined here the concept of a **total** function. For partial function we assume an element $\perp$, the undefined element, which is not in B (the range of the function).

f is a **partial function** from A into B iff f: A $\rightarrow$ B $\cup$ $\{\perp\}$
A partial function from A into B is a total function from A into B $\cup$ $\{\perp\}$.

For partial functions f,g: A $\rightarrow$ B we define:

$$f \subseteq g \text{ iff for every } a \in A\text{: if } f(a) \in B \text{ then } f(a) = g(a)$$

This means that g assigns the same values that f assigns a value to, but possibly assigns values to more arguments than f does.

Our functions will generally, but not always, be total functions. We are not going to be very precise about the distinction.

Special functions
Let f be a function from A into B.

f is an **injection, one-one,** iff for every $a_1,a_2 \in A$: if $a_1 \neq a_2$ then $f(a_1) \neq f(a_2)$
f is a **surjection, onto,** iff **ran**(f) = B
f is a **bijection** iff f is an injection and a surjection.
The **identity function:** $\mathbf{id}_A$: $A \rightarrow A$ is the function such that for every $a \in A$: $\mathbf{id}_A(a) = a$

Let $B \subseteq A$

The **characteristic function** of B (in A): $\mathbf{ch}_B$: $A \rightarrow \{0,1\}$ is the function such that
for every $a \in A$: $\mathbf{ch}_B(a)=1$ iff $a \in B$

The **constant function from** A **into** B **on** element $b \in B$ is the function $\mathbf{c}_b$: $A \rightarrow B$ such that:
for every $a \in A$: $\mathbf{c}_b(a) = b$

Let f: $A \rightarrow B$ be a one-one function. Then the converse relation f^{-1} is itself a bijection from **ran**(f) into A. In this case we call f^{-1} the **inverse function**.

Let f:$A \rightarrow B$ and g: $B \rightarrow C$ be functions.

the **composition** of g and f, $g \circ f$: $A \rightarrow C$, is the function that maps every $a \in A$ onto g(f(a)).
We visualize this in λ-notation:

$g \circ f = \lambda a.g(f(a))$ the function that maps every a onto g(f(a))

Composition in number phrases

number: *three* → 3
number relation: *at least* → ≥
$\lambda m \lambda n.n \geq m$
the relation that holds between n and m if $n \geq m$
the function that maps n and m onto truth value 1 iff $n \geq m$
at most → ≤ *exactly* → = *more than* → > *less than* → <

2 place relations between numbers = 2 place functions from numbers into truth values:
functions f: $N \times N \rightarrow \{0,1\}$

Semantics of application: n-place relation + argument → n–1-place relation
R^n + arg → $(R^n(arg))$
at least three → ≥(3)
$\lambda n.n \geq 3$
the property that n has if $n \geq 3$
the function that maps n onto truth value 1 iff $n \geq 3$
the set of numbers n such that $n \geq 3$

at most three → $\lambda n.n \leq 3$ *exactly three* → $\lambda n.n = 3$
more than three → $\lambda n.n > 3$ *less than three* → $\lambda n.n < 3$

1 place predicates of numbers = 1 place functions from numbers into truth values = functions $f: N \rightarrow \{0,1\}$

cardinality function: card = $\lambda x.|x|$
The function that maps singular or plural objects onto their cardinality.
Let D be the domain of objects.

card: $D \rightarrow N$ function from objects to numbers

Nouns: *cats* is intepreted as CATS: $D \rightarrow \{0,1\}$
the function that maps an object onto 1 iff it is a cat or a plurality of cats

at least three	+	*cats*	→	*at least three cats*
$\lambda n.n \geq 3$		CATS		?
$N \rightarrow \{0,1\}$		$D \rightarrow \{0,1\}$		$D \rightarrow \{0,1\}$

Composition: 1-place predicate of numbers ∘ **card** → 1-place predicate of objects
$N \rightarrow \{0,1\}$ $D \rightarrow N$ $D \rightarrow \{0,1\}$

Composition: *at least three* + [card] → *at least three*
$\lambda n.n \geq 3 \circ \lambda x.|x| \rightarrow \lambda n.n \geq 3 \circ \lambda x.|x|$
$N \rightarrow \{0,1\}$ $D \rightarrow N$ $D \rightarrow \{0,1\}$

$$\lambda n.n \geq 3 \circ \lambda x.|x| = \boldsymbol{\lambda z.(}\lambda n.n \geq 3\boldsymbol{(}\lambda x.|x|\boldsymbol{(z)))} =$$
$$\boldsymbol{\lambda z.(}\lambda n.n \geq 3\boldsymbol{(|z|))} =$$
$$\boldsymbol{\lambda z.(|z|} \geq 3\boldsymbol{)}$$

at least three → $\lambda x.|x| \geq 3$
the property that x has if $|x| \geq 3$
the function that maps x onto truth value 1 iff $|x| \geq 3$
the set of objects x such that $|x| \geq 3$
the set of pluralities that are sums of at least 3 singularities

at most three → $\lambda x.|x| \leq 3$ *exactly three* → $\lambda x.|x| = 3$
more than three → $\lambda x.|x| > 3$ *less than three* → $\lambda x.|x| < 3$

at least three	+	*cats*	→	*at least three cats*
$\lambda x.|x| \geq 3$		CATS		$\lambda x.|x| \geq 3 \cap$ CATS
				$\lambda x.CATS(x) \wedge |x| \geq 3$
$D \rightarrow \{0,1\}$		$D \rightarrow \{0,1\}$		$D \rightarrow \{0,1\}$

$\lambda x.CATS(x) \wedge |x| \geq 3$
The set of cat-pluralities that are sums of at least three cat-singularities.
(see the plurality theory discussed later in this class).

Moral:
Semantics of application to arguments:

n-place relation + argument → n–1-place relation

R^n + arg → $(R^n(arg))$

Semantics of intersective adjectives:

adjective + NP → NP

ADJ N → ADJ ∩ N

Composition with card

1-place predicate of numbers ∘ **card** → 1-place predicate of objects

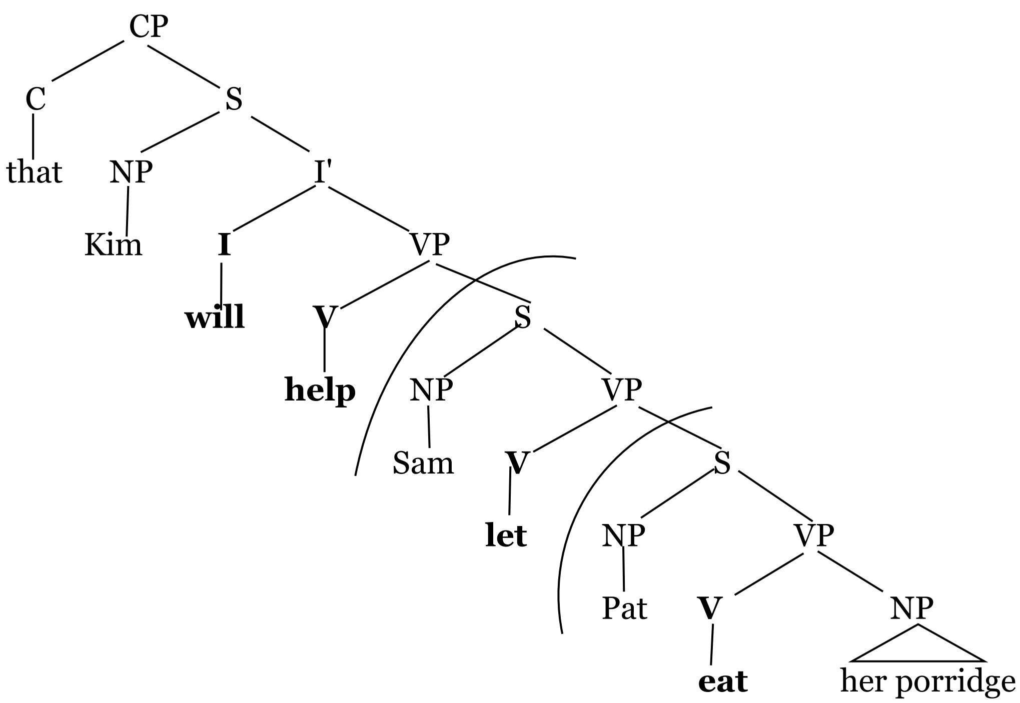
N → {0,1} D → N D → {0,1}

Composition with card is a systemsatic principle that shifts the interpretation of *at least three* as a set of numbers to the corresponding interpretation of *at least three* as a set of objects.

Composition in the verb cluster in Dutch and German.

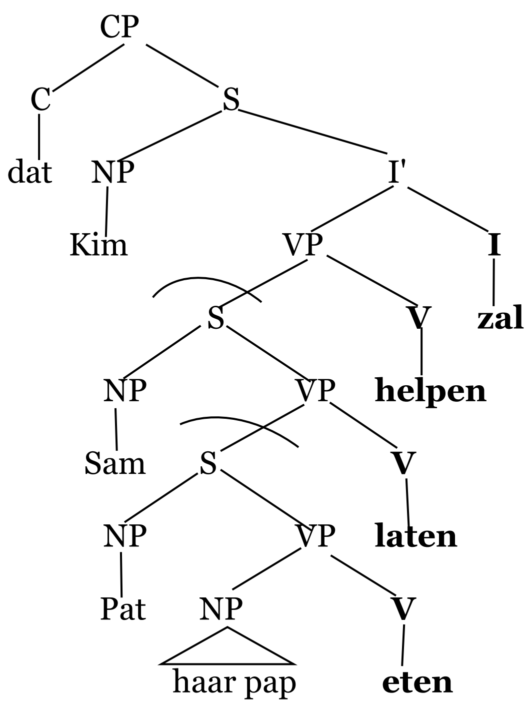
(1) That Kim will *help* Sam *let* Pat eat her porridge.

A standard assumption is that cases like (1) get a **small clause analysis**.

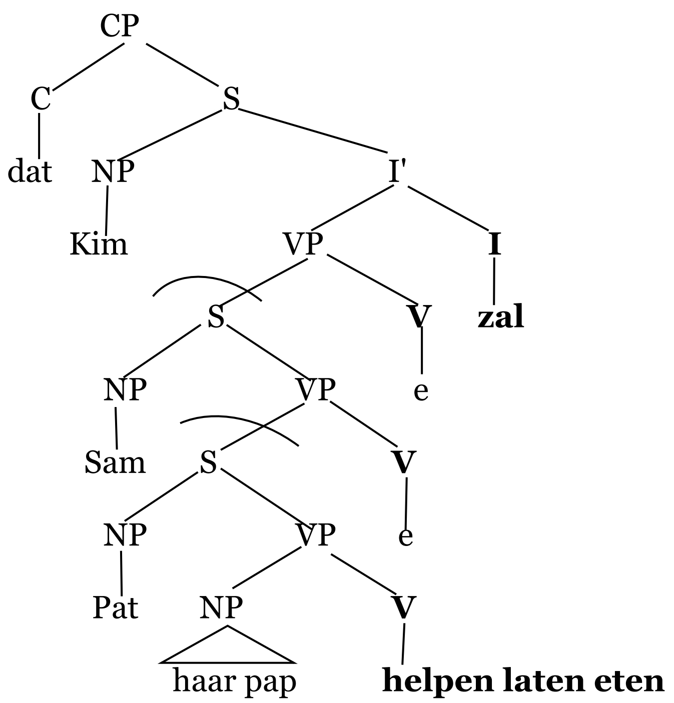


verbs: help, let, see, hear, make…

In Dutch and German V and I, are assumed to be on the right. The same class of verbs take bare infinitives. This means that, following the English analysis, we would expect to find for Dutch:



This is not what we find. What we find is arguably more like (arguments in my class notes Formal Languages):



German: Ihr Brei **essen lassen helfen** wird

Semantic assumption: transitive verbs

eat → $\lambda y\lambda x.\mathrm{EAT}(x,y)$

The relation that x and y stand in if x eats y =
The function that maps x and y onto 1 iff x eats y =
The function that maps y onto the one place function $\lambda x.\mathrm{EAT}(x,y)$,
which is the function that maps x onto 1 iff x eats y.

Semantics of application to arguments:
An n-place relation applies to an argument to form an n–1 place relation

Semantic assumption: let, help, see…. are semantically 3 place relations:

let → $\lambda P\lambda y\lambda x.\mathrm{LET}(x,P(y))$ (call it LET)

3-place relation between two individuals and a property

"x lets it be brought about that y has P"

help → $\lambda P\lambda y\lambda x.\mathrm{HELP}(x,P(y))$ (call it HELP)

"x helps it be brought about that y has P"

similarly, *see, hear, make* (the latter only in English)…

Semantics of English: just use application at all stages

Semantics of Dutch and German:
The verb cluster is a complex verb. Inside category V the semantics uses function composition as its general tool.

Generalized composition: $g \circ f = \lambda x_n \ldots \lambda x_1.g(f(x_1,\ldots x_n))$

Hence: *helpen laten eten* → HELP ∘ LET ∘ EAT

Step 1: LET ∘ EAT =

$\lambda P\lambda y\lambda x.\mathrm{LET}(x,P(y)) \circ \lambda y\lambda x.\mathrm{EAT}(x,y) =$

$\lambda \mathbf{z}.\mathbf{(}\lambda P\lambda y\lambda x.\mathrm{LET}(x,P(y))\ \mathbf{(}\lambda y\lambda x.\mathrm{EAT}(x,y)\mathbf{(z))}\ \mathbf{)} =$

This expressions simplifies with **λ-conversion**: (see class notes on Advanced semantics)

Step 1: $\lambda z\lambda P\lambda y\lambda x.\mathrm{LET}(x,P(y))\ (\boldsymbol{\lambda y}\lambda x.\mathrm{EAT}(x,\mathbf{y})\mathbf{(z)}) =$

$\lambda z\lambda P\lambda y\lambda x.\mathrm{LET}(x,P(y))\ (\lambda x.\mathrm{EAT}(x,\mathbf{z}))$

Step 2: $\lambda z\lambda \mathbf{P}\lambda y\lambda x.\mathrm{LET}(x,\mathbf{P}(y))\ (\boldsymbol{\lambda x}.\mathbf{EAT(x,z)}) =$

$\lambda z\lambda y\lambda x.\mathrm{LET}(x,\ \boldsymbol{\lambda x}.\mathbf{EAT(x,z)}\ (y))$

Step 3: $\lambda z\lambda y\lambda x.\mathrm{LET}(x,\ \boldsymbol{\lambda x}.\mathrm{EAT}(\mathbf{x},z)\ (\mathbf{y})) =$

$\lambda z\lambda y\lambda x.\mathrm{LET}(x,\ \mathrm{EAT}(\mathbf{y},z))$

LET ∘ EAT = $\lambda z\lambda y\lambda x.\mathrm{LET}(x,\ \mathrm{EAT}(y,z))$ 3-place relation

The relation that x, y and z stand in iff x lets it be brought about that y eats z

HELP ∘ LET ∘ EAT = $\lambda P\lambda y\lambda x.\mathrm{HELP}(x,P(y)) \circ \lambda z\lambda y\lambda x.\mathrm{LET}(x,\ \mathrm{EAT}(y,z))$

λPλyλx.HELP(x,P(y)) ∘ λzλyλx.LET(x, EAT(y,z)) =
λuλzλPλyλx.HELP(x,P(y)) (λzλyλx.LET(x, EAT(y,z))**(u,z)**) =

Notice: Generalized composition requires that we apply the three place relation λzλyλx.LET(x, EAT(y,z)) to **two** variables, which gives a one place property. This is because HELP is a function that must apply first to a one place property. After that, generalized composition abstracts over those two variables.

We simplify with λ-conversion:

Step 1: λzλu.λPλyλx.HELP(x,P(y)) (**λzλy**λx.LET(x, EAT(y,z))**(u,z)**) =
λzλuλPλyλx.HELP(x,P(y)) (λx.LET(x, EAT(**u,z**))

Step 2: λzλu**λP**λyλx.HELP(x,**P**(y)) (**λx.LET(x, EAT(u,z))**) =
λzλuλyλx.HELP(x, (**λx.LET(x, EAT(u,z))**(y))

Step 3: λzλuλyλx.HELP(x, (**λx**.LET(**x**, EAT(u,z))(**y**)) =
λzλuλyλx.HELP(x, LET(**y**, EAT(u,z)))

helpen laten eten → λzλuλyλx.HELP(x, LET(y, EAT(u,z)))
four place relation:
The relation that holds between x, y, u and z iff
x helps it be brought about that y lets it be brought about that u eats z

Application: R^n + arg = R^{n-1}

pap helpen laten eten = λuλyλx.HELP(x, LET(y, EAT(u,**porridge**)))

The three place relation that holds between x, y and u if x helps it be brought about that y lets it be brought about that u eats porridge.

Pat pap helpen laten eten = λyλx.HELP(x, LET(y, EAT(**Pat**,**porridge**)))

The two place relation that holds between x and y if x helps it be brought about that y lets it be brought about that Pat eats porridge.

Sam Pat pap helpen laten eten = λx.HELP(x, LET(**Sam**, EAT(**Pat**,**porridge**)))

The one place property that x has if x helps it be brought about that San lets it be brought about that that Pat eats porridge

Kim Sam Pat pap helpen laten eten = HELP(**Kim**, LET(**Sam**, EAT(**Pat**,**porridge**)))

The null-place relation (= statement) that Kim helps it be brought about that San lets it be brought about that that Pat eats porridge

Crucial property of composition:

Let Z be a **three place relation** between **two objects** and a **property**
Let R be an **n-place relation** between **objects**
Then Z ∘ R is an **n+1 place relation** between **objects**

Thus composition gives languages the capacity to create n-place relations.
This is what seems to happen with serial verbs.

An **n-place operation on** A is a function from A^n into A.
A one-place operation which is a bijection is called a **permutation**.
n-place operations are called **finitary** operations if n is finite.
Operations can also be infinitary. To choose a terminology that is suited for finite and infinite sets, I will use the expression **complete** (the terminology will be explained later):

A **complete operation** on A is a function from pow(A) into A.
In fact, it will be useful to introduce a partial variant of this notion:
A **complete$^+$ operation** on A is a partial function f from pow(A) into A, which is total,
except that $f(\emptyset) = \bot$

Let $B \subseteq A$ and let $R \subseteq A^n$ and let $f: A^n \to C$
The **restriction** of R to B, $R \upharpoonright B = \{<a_1,\ldots,a_n> \in R: a_1,\ldots,a_n \in B\}$
The **restriction** of f to B, $f \upharpoonright B = \{<b_1,\ldots,b_n : f(<b_1,\ldots,b_n>)>: b_1,\ldots,b_n \in B\}$

Structures (of finite type)

A **structure** is a quintuple **A** = $<A, R_A, O_A, S_A, \tau_A>$ where:

1. A is a non-empty set
2. R_A is a finite set of relations on A
3. O_A is a finite set of operations on A
4. S_A is a finite set of special elements of A
5. τ_A is the union of an enumeration of R_A, an enumeration of O_A and an enumeration of S_A

τ is an enumeration of finite set X iff τ is a bijection between X and an initial segment of $\mathbf{N}^+$.
(An initial segment of N^+ is a set $\{1,2,\ldots n\}$, for some $n \in N$).

Let **A** and **B** be structures. Let $r_A \in R_A$ and $r_B \in R_B$.

r_A and r_B are **corresponding relations** iff $\tau_A(r_A) = \tau_B(r_B)$
The same for operations and special elements.

τ_A **is a subtype of** τ_B, $\tau_A \subseteq \tau_B$, iff all relations, functions, special elements in **A** have
corresponding relations, functions, and special elements in **B of the same arity**.

(i.e. if r_A is an n-place relation in R_A, then there is a corresponding n-place relation in R_B).

A and **B** have the same type, $\tau_A = \tau_B$ iff $\tau_A \subseteq \tau_B$ and $\tau_B \subseteq \tau_A$

The central relations between structures are standardly defined for structures of **the same type,** sometimes for structures where one is of a subtype of the other.
Universal algebra defines its notions at this level of generality. Algebra and logic (and semantics) are usually concerned with much more specific notions. You get these by putting restrictions on the structures **and** restrictions on the notion of *corresponding relation/operation/special element*. Thus in a structure which is a partial order you assume a two-place relation which is transitive. To be of the same type, it is not enough to be a structure which has a corresponding two-place relation, but the relation itself must be transitive as well. In other words, the relevant notions of structure preservation that we will be concerned with below compare partial orders not just with structures of the same type, but with partial orders of the same type.

Various of the sets R_A, O_A, S_A can be empty.

A **relational structure** is a structure $\mathbf{A} = <A, R_A, \tau_A>$
An **algebra** is a structure $\mathbf{A} = <A, O_A, \tau_A>$
A **relational algebra** is a structure $\mathbf{A} = <A, R_A, O_A, \tau_A>$

There is no special termimology for structures with special elements. In fact, with a characteristic bit of mathematician’s humour, special elements are standardly subsumed under operations, as 0-place operations.

In practice, we suppress the reference to the type in the notion of structure, and for ease, we will define the relevant notions for *token structures* which have one two-place relation R, one two-place operation and one special element:

$\mathbf{A} = <A, R_A, *_A, s_A>$ $\quad$ $\mathbf{B} = <B, R_B, *_B, s_B>$

The use of the same symbols R, * and s indicates corresponding relations, operations and special elements. I will use infix notation for $*_A$: and write $*_A(a_1,a_2)$ as: $(a_1 *_A a_2)$.

Thus, $\mathbf{A_1} = <A, R_A>$ is a relational structure
$\mathbf{A_2} = <A, *_A>$ is an algebra
$\mathbf{A_3} = <A, R_A, *_A>$ is a relational algebra
$\mathbf{A_4} = <A, R_A, *_A, s_A>$ is also a relational algebra.

Important: Technically the structures $\mathbf{A_1} \ldots \mathbf{A_4}$ count as distinct structures, even though they are the same set, structured by the same relations and operations. But they will count as different for the sake of structure preservation relations that we discuss now.

So in what follows we are concerned with structures **of the same type**.

Examples of special relations: Partial orders

Let R be a two place relation on A.
R is **reflexive** iff for all $a \in A$: R(a,a)
R is **irreflexive** iff for no $a \in A$: R(a,a)
R is **transitive** iff for for all $a,b,c \in A$: if R(a,b) and R(b,c) then R(a,c)
R is **intransitive** iff for all $a,b,c \in A$: if R(a,b) and R(b,c) then $\neg$R(a,c)
R is **symmetric** iff for all $a,b \in A$: if R(a,b) then R(b,a)
R is **asymmetric** iff for all $a,b \in A$: if R(a,b) then $\neg$R(b,a)
R is **antisymmetric** iff for all $a,b \in A$: if R(a,b) and R(b,a) then a=b

R is **connected** iff for all $a,b \in A$: R(a,b) or R(b,a) or a=b

R is a **preorder** iff R is reflexive and transitive
R is a **partial order** iff R is reflexive, transitive and antisymmetric
R is a **strict partial order** iff R is irreflexive, transitive and asymmetric

R is a (strict) **total or linear order** iff R is a connected (strict) partial order

R is an **equivalence relation** iff R is reflective, transitive and symmetric.

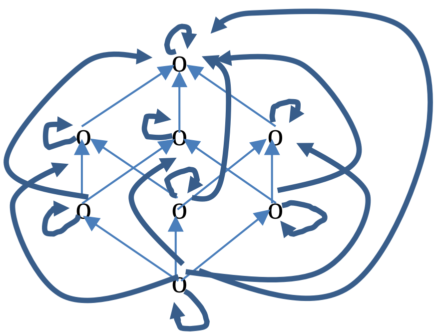
Fact: every partial order determines a strict partial order and vice verse.
Namely:
-Let $\sqsubseteq$ be a partial order.
Define: $x \sqsubset y$ iff $x \sqsubseteq y \wedge x \neq y$
Then $\sqsubset$ is a strict partial order.
-Let $\sqsubset$ be a strict partial order.
Define: $x \sqsubseteq y$ iff $x \sqsubset y \vee x = y$
Then $\sqsubseteq$ is a partial order.

This terminology extends to structures:

Structure **A** = $\langle A, \sqsubseteq \rangle$ is a partial order iff $\sqsubseteq$ is a partial order.

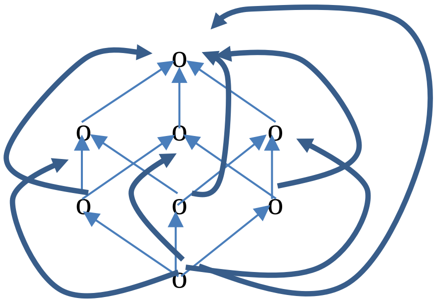
We simplify the graphs of partial orders in the following way:

Consider the following partial order.

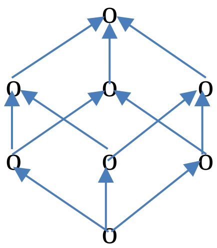


We apply the following conventions.

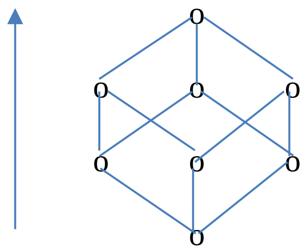
1. We do not distinguish in the graph between strict partial orders and partial orders, and hence do not write the reflexivity arrows:



2, We take transitivity to be understood in the graph: when an arrow goes from a to b and from b to c, we do not write the arrow from a to c: we assume it is there:



3. Since in a partial order all arrows go in the same direction, we take in the graph the direction as understood. This means we do not have to write the arrow heads:

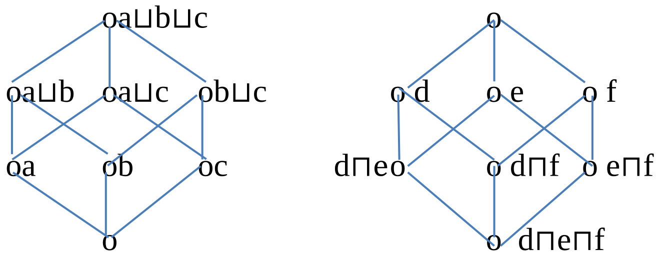


Examples of special operations: join and meet

Let **A =** <A, ⊑> be a partial order, let a,b ∈ A.

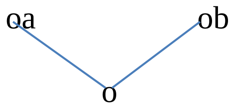
The **join** of a an b under ⊑, a ⊔ b, is the minimal element of A such that:
a ⊑ a ⊔ b and b ⊑ a ⊔ b
The **meet** of a an b under ⊑, a ⊓ b, is the maximal element of A such that:
a ⊓ b ⊑ a and a ⊓ b ⊑ b

A

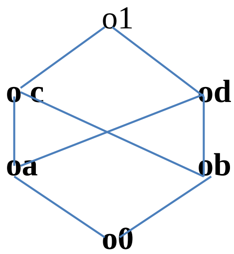


The above example is an example of a partial order in which join and meet are defined for every two elements. This is not generally the case for partial orders:

B



C

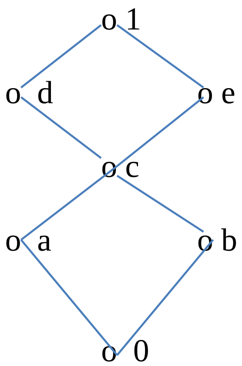


In **B** there is no minimal element in B that both a and b are part of, so a⊔b is not defined.
In **C** there are three elements that are part of both c and d: a, b and 0, but there is no element that is maximally such, so c⊓d is not defined. Similarly, a⊔b is not defined, because no element is "the mimimal element that both a and b are part of".

In other words:

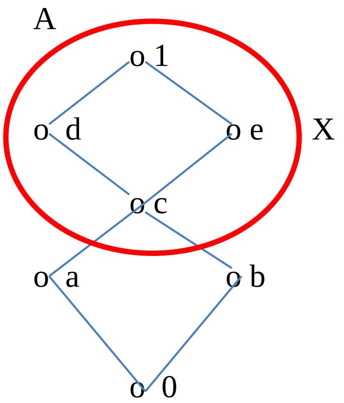
Look at the following partial order (with ⊑ going up):

A

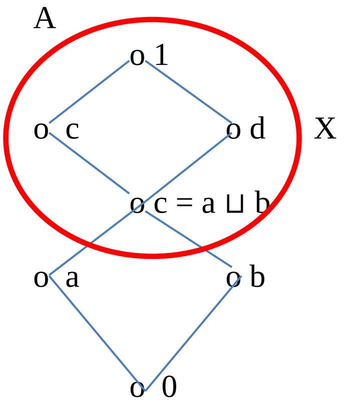


We want to know what the join is of a and b: a ⊔ b.

Step 1: We look at the set: $\{x \in A: a \sqsubseteq x \text{ and } b \sqsubseteq x\}$
This is the set X marked in red:

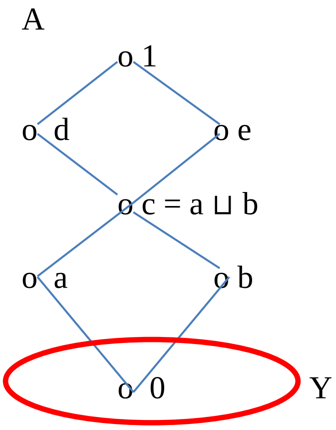


Step 2: The join of a and b, $a \sqcup b$ is the minimum of X:



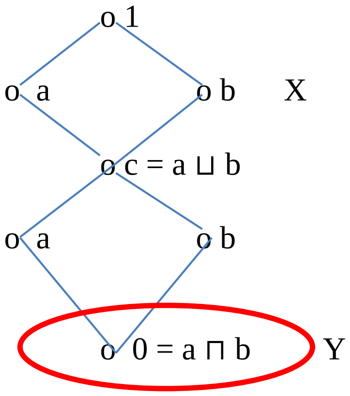
This is the smallest element of A of which both a and b are part.

We want to know what the meet is of a and b: $a \sqcap b$.
Step 1: We look at the set: $\{x \in A: x \sqsubseteq a \text{ and } x \sqsubseteq b\}$
This is the set Y marked in red:



Step 2: The meet of a and b, $a \sqcap b$ is the maximum of X:

A



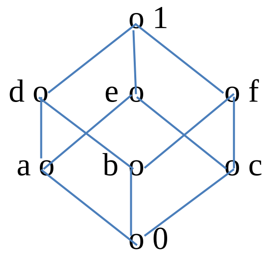
The meet of a and b, a ⊓ b is the biggest element that is part both of a and of b.

a ⊔ b is only defined if the set {x ∈ A: a ⊑ x and b ⊑ x} has a minimum.
a ⊓ b is only defined if the set {x ∈ A: a ⊑ x and b ⊑ x} has a maximum.

It is useful practice to take the above picture, or some of the other ones in the text, and determine for any two elements what are their join and meet.

While the notions of join and meet are defined here for partial orders, they can be defined independently as two place operations. Thus, the operations of join and meet on **D** can be specified by the tables given:

D



⊔	0	a	b	c	d	e	f	1
0	0	a	b	c	d	e	f	1
a	a	a	d	e	d	e	1	1
b	b	d	b	f	d	1	f	1
c	c	e	f	c	1	e	f	1
d	d	d	d	1	d	1	1	1
e	e	e	1	e	1	e	1	1
f	f	1	f	f	1	1	f	1
1	1	1	1	1	1	1	1	1

⊓	0	a	b	c	d	e	f	1
0	0	0	0	0	0	0	0	0
a	0	a	0	0	a	a	0	a
b	0	0	b	0	b	0	b	b
c	0	0	0	c	0	c	c	c
d	0	a	b	0	d	a	b	d
e	0	a	0	c	a	e	c	e
f	0	0	b	c	b	c	f	f
1	0	a	b	c	d	e	f	1

Special elements: minimum 0 and maximum 1

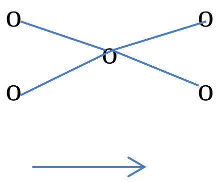
Let **A** = <A,⊑> be a partial order, a ∈ A

a is a **minimal element in A** iff for no b ∈ A: b ⊏ a
a is a **maximal element in A** iff for no b ∈ A: a ⊏ b

a is **the minimum of A** iff for all b ∈ A: a ⊑ b
a is **the maximum of A** iff for all b ∈ A: b ⊑ a

If **A** has a minimum, we call the minimum 0_A (or just 0)
If **A** has a maximum, we call the maximum, 1_A (or just 1)

A partial order can have more than one minimal element. The following structure with orientation from left to right has two minimal elements and two maximal elements. It has neither a minimum, nor a maximum.



The use of the definite article in minimum invites the following observation:

Fact: if partial order **A** has a mimium then it has a unique minimum.

Proof: if m_1 and m_2 are both minimums of **A** then $m_1 \sqsubseteq m_2$ and $m_2 \sqsubseteq m_1$, and by antisymmetry $m_1 = m_2$.

With this we see:

D $= \langle D,\sqsubseteq\rangle$ is a relational structure (a partial order)
D $= \langle D,\sqcap,\sqcup\rangle$ is an algebra with operations join and meet (a lattice)
D $= \langle D,0,1\rangle$ is a structure with special elements 0 and 1
D $= \langle D,\sqsubseteq,\sqcap,\sqcup\rangle$ is a relational algebra
D $= \langle D,\sqsubseteq,\sqcap,\sqcup,0,1\rangle$ is a relational algebra with special elements 0 and 1

Substructures.
Let **A** $= \langle A, R_A, *_A, s_A\rangle$ and **B** $= \langle B, R_B, *_B, s_B\rangle$ be structures of the same type.

A $\subseteq$ **B**, **A** is a **substructure** of **B** iff

1. $A \subseteq B$
2. $R_A = R_B \restriction A$
3. $*_A = *_B \restriction A$
4. $s_A = s_B$

Important: $*_A = *_B \restriction A$
for all $b_1,b_2 \in B$: $(b_1 *_B b_2) \in B$
$*_A$ is defined only for pairs of objects $b_1,b_2 \in A$, but assigns the same value to these as $*_B$ does.
hence for $b_1,b_2 \in A$: $(b_1 *_A b_2) \in B$.
How do we know that in fact for $b_1,b_2 \in A$: $(b_1 *_A b_2) \in A$?
Answer: by the definition of substructure.
if **A** is a substructure of **B** then **A** is a structure.
If **A** is a structure then **A** is closed under $*_A$, hence A is closed under $*_B \restriction A$.
And that means precisely that $b_1,b_2 \in A$: $(b_1 *_A b_2) \in A$.

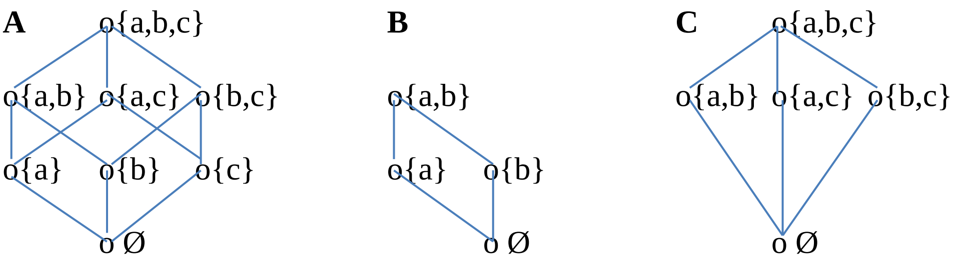
Given this, a substructure of **B** should be distinguished from a structure of the same type as **B** on a subset of **B**.

For instance, look at the following structures:

A $= \langle$**pow**$(\{a,b,c\}), \cap\rangle$
B $= \langle$**pow**$(\{a,b\}), \cap\rangle$
C $= \langle$**pow**$^+(\{a,b,c\}),\cap^+\rangle$

where **pow**$(\{a,b,c\})^+ =$ **pow**$(\{a,b,c\}) - \{\{a\},\{b\},\{c\}\}$
and where $\cap^+$ is defined as:

$$X \cap^+ Y = \begin{cases} X \cap Y & \text{if } X \cap Y \in \textbf{pow}^+(\{a,b,c\}) \\ \emptyset & \text{otherwise} \end{cases}$$



B $\subseteq$ **A**
C is not a substructure of **A** but a structure of the same type on a subset of **A**.
The reason is that the operations of **A** are not preserved in **C**.

With the same pictures look at the structures:
A' = <**pow**({a,b,c}), $\cap$, Ø, {a,b,c} > and **B'** = <**pow**({a,b}), $\cap$, {a,b}, Ø >
While **B** is a substructure of **A**, **B'** is not a substructure of **B**, since the special elements are not preserved, in particular, {a,b,c} is a special element in **A**, but the corresponding special element in **B** is {a,b}. This violates the last clause of *substructure*. So **B'** is a structure of the same type as **A'** on a subset of **pow**({a,b,c}), not a substructure.

Let **B** = <B, R_B, $*_B$, s_B> and let A $\subseteq$ B

The **restriction of B** to A, **B**↾A is defined as:

B↾A = <A, R_B↾A, $*_B$↾A, s_B>

The restriction of **B** to A is only defined if A is closed under $*_B$ and $s_B \in$ A.

Fact: if **B**↾A is defined then **B**↾A $\subseteq$ **B**

Of course you **can** generalize the notion of substructure to a relation between two structures **A** and **B** where **A** is a structure of a **subtype** of **B**, and that is a useful notion.
This would allow a structure <A, $\cap$ > to be a substructure of <A, $\cap$, $\cup$ > and also of <B, $\cap$, $\cup$ > (if <A, $\cap$, $\cup$ > $\subseteq$ <B, $\cap$, $\cup$ >).

Homomorphisms
Let **A** = <A,R_A, $*_A$, s_A> and **B** = <B,R_B, $*_B$, s_B> be structures of the same type.

h is a **homorphism from A into B** iff

1. h is a function from A into B
2. h preserves the structure:
    1. for all $a_1,a_2 \in$ A: if $R_A(a_1,a_2)$ then $R_B(h(a_1),h(a_2))$
    2. for all $a_1,a_2 \in$ A: $h(a_1 *_A a_2) = h(a_1) *_B h(a_2)$
    3. $h(s_A) = s_B$

Let h be a homomorphism from **A** into **B**.

h is an **injective homomorphism** iff h is an injection
h is a **surjective homomorphism** iff h is a surjection
h is a **bijective homomorphism** iff h is a bijection

h is a **strong homomorphism** iff h **anti-preserves** R_A:
for all $a_1,a_2 \in A$: if $R_B(h(a_1),h(a_2))$ then $R_A(a_1,a_2)$

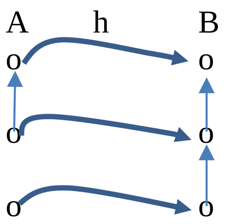
This means that for a strong homomorphism h we have:
for all $a_1,a_2 \in A$: $R_A(a_1,a_2)$ iff $R_B(h(a_1),h(a_2))$

h is an **embedding** iff h is a strong injective homomorphism
h is an **epimorphism** iff h is a strong surjective homomorphism
h is an **isomorphism** iff h is a strong bijective homomorphism

A and **B** are **isomorphic** iff there is an isomorphism between **A** and **B**.

h is an **automorphism** on **A** iff h is a homomorphism from **A** into **A**.

The notion of strength has to do with the preservation of the relation. For relational structures a bijective homomorphism isn't yet an isomorphism.
Example:



h is a bijective homomorphism from A into B, a one-one function such that for every $a_1,a_2 \in A$: if $R_A(a_1,a_2)$ then $R_B(h(a_1),h(a_2))$. But h is not an isomorphism, because the relation is not anti-preserved: it is not the case that for every $a_1,a_2 \in A$: if $R_B(h(a_1),h(a_2))$ then $R_A(a_1,a_2)$.
It is easy to see that if our structures are **algebras** (understood as structures without special relations), then the notion of embedding and injective homomorphism coincide, and so do the notions of epimorphism and surjective homomorphism, and also isomorphism and bijective homomorphism.

Let h be a homomorphism from **A** into **B.**

The **homomorphic image of A in B under** h is the structure:
$h(\mathbf{A}) = \langle h(A), R_{h(A)}, *_{h(a)}, s_{h(A)} \rangle$ where:

1. $h(A) = \{h(a): a \in A\}$
2. $R_{h(A)} = \{\langle h(a_1),h(a_2)\rangle: a_1,a_2 \in A \text{ and } R_B(h(a_1),h(a_2))\}$
3. $*_{h(A)}$ is the unique operation on h(A) such that
for all $a_1,a_2 \in A$: $h(a_1) *_{h(A)} h(a_2) = h(a_1 *_A a_2)$
4. $s_{h(A)} = h(s_A)$

Fact: $h(\mathbf{A}) \subseteq \mathbf{B}$, in fact, $h(\mathbf{A}) = \mathbf{B}{\restriction}h(A)$

If h is an epimorphism from **A** into **B** then $\mathbf{B} = h(\mathbf{A})$
If h is an embedding, then $h(\mathbf{A})$ is called the **isomorphic image of A inside B**.
If h is an isomorphism the obviously **B is** the isomorphic image of **A**.

Partitions

We defined an equivalence relation as a reflexive, transitive and symmetric relation.

Let A be a set.
Covers and partitions on A are sets of subsets of A. The subsets of a cover or a partition are called the **blocks**, or the cells of the cover or partition.

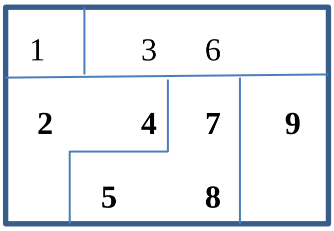
A cover of A is a set C_A such that:

1. $C_A \subseteq \mathbf{pow}(A)$ and $\varnothing \notin C_A$ — C_A is a set of non-empty subsets of A
2. $A = \cup\{B: B \in C_A\}$ — C_A covers A

A partition of A is a set P_A such that:

1. P_A is a cover of A.
2. for all $B_1,B_2 \in P_A$: if $B_1 \neq B_2$ then $B_1 \cap B_2 = \varnothing$
   The blocks of P_A do not overlap

A partition of A is a cover of A where the blocks form a minimal cover of A.



There is a natural relation between partitions and equivalence relations:

Let A be a set and $\approx$ an equivalence relation on A and $a \in A$.

The **equivalence class of** a **under** $\approx$, $[a]_\approx$, is:

$[a]_\approx = \{b \in A: b \approx a\}$

a is called the *representative element* of equivalence class $[a]_\approx$

$[A]_\approx = \{[a]_\approx: a \in A\}$

Fact 1: If $\approx$ is an equivalence relation of A then $[A]_\approx$ is a partition of A
Proof:
We prove the following four lemmas:

Lemma 1. if $b \in [a]_\approx$ then $[a]_\approx = [b]_\approx$
-If $b \in [a]_\approx$ then $b \approx a$, and hence (by symmetry) $a \approx b$.
Now:
-If $x \in [a]_\approx$ then $x \approx a$. Since $a \approx b$, (by transitivity) $x \approx b$. Then $x \in [b]_\approx$
-If $x \in [b]_\approx$, then $x \approx b$. Since $b \approx a$ (by transitivity) $x \approx a$. Hence $x \in [a]_\approx$

Lemma 2. For every $B \in [A]_\approx$, $B \neq \emptyset$
If $B \in [A]_\approx$, then for some $a \in A$: $B = [a]_\approx$. Since $\approx$ is reflexive $a \in [a]_\approx$. Hence $B \neq \emptyset$.

Lemma 3. if $[a]_\approx \neq [b]_\approx$ then $[a]_\approx \cap [b]_\approx = \emptyset$
Assume $[a]_\approx \cap [b]_\approx \neq \emptyset$, say, $x \in [a]_\approx \cap [b]_\approx$
Then $x \in [a]_\approx$ and $x \in [b]_\approx$. Then, by Lemma 1, $[a]_\approx = [x]_\approx = [b]_\approx$.

Lemma 4. $\cup[A]_\approx = A$
-Let $a \in A$. Then (by reflexivity) $a \in [a]_\approx$. Since $[a]_\approx \in [A]_\approx$, $a \in \cup[A]_\approx$
-Let $a \in \cup[A]_\approx$. Then for some $B \in [A]_\approx$, $a \in B$. Since $B \subseteq A$, $a \in A$.
□

For partition P_A we define:

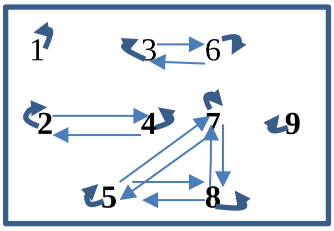
$$a_1 \approx_P a_2 \text{ iff } \exists B \in P_A\text{: } a_1 \in B \text{ and } a_2 \in B$$

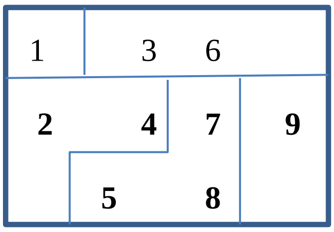
The following two facts are easy to check.

Fact 2. If P_A is a partition of A, then $\approx_P$ is an equivalence relation on A

Fact 3: If $\approx$ is an equivalence relation and $P = [A]_\approx$, then $\approx_P = \approx$

This means that indeed equivalence relations and partitions are two sides of the same coin.





We define language L.
L has one *individual constant*: s
L has one *two-place relational constant*: R
L has one *two-place functional constant*: *

VAR = $\{x_1, x_2, x_3, \ldots\}$ — a countable set of *individual variables*.

$TERM_L$ is the smallest set such that: — *terms* of L

1. $s \in TERM_L$
2. $VAR \subseteq TERM_L$
3. If $t_1, t_2 \in TERM_L$ then $(t_1 * t_2) \in TERM_L$

$ATFORM_L$ is the smallest set such that: — *atomic formulas* of L

1. If $t_1, t_2 \in TERM_L$ then $R(t_1, t_2) \in ATFORM_L$
2. If $t_1, t_2 \in TERM_L$ then $(t_1 = t_2) \in ATFORM_L$

$FORM_L$ is the smallest set such that: — *formulas* of L

1. $ATFORM_L \subseteq FORM_L$
2. If $\varphi \in FORM_L$ then $\neg\varphi \in FORM_L$
3. If $\varphi, \psi \in FORM_L$ then $(\varphi \wedge \psi), (\varphi \vee \varphi) \in FORM_L$
4. If $x \in VAR$ and $\varphi \in FORM_L$ then $\exists x\varphi, \forall x\varphi \in FORM_L$

$POSFORM_L$ is the smallest set such that: — *positive formulas* of L

1. $ATFORM_L \subseteq FORM_L$
3. If $\varphi, \psi \in FORM_L$ then $(\varphi \wedge \psi), (\varphi \vee \varphi) \in FORM_L$
4. If $x \in VAR$ and $\varphi \in FORM_L$ then $\exists x\varphi, \forall x\varphi \in FORM_L$

Positive formulas are formed without use of negation.

A structure for L is a structure $\mathbf{A} = \langle A, R_A, *_A, s_A \rangle$
An assignment function on **A** is a function $g: VAR \rightarrow A$
g_x^a = the assignment that differs at most from g in that g(x)=a
i.e. $g_x^a(y) = = g(y)$ if $y \neq x$
$g_x^a(x) = a$

Interpretation: We define $[\![\alpha]\!]_{\mathbf{A},g}$ the interpretation of α in **A** relative to g:

Constants:	$[\![s]\!]_{\mathbf{A},g} = s_A$
	$[\![R]\!]_{\mathbf{A},g} = R_A$
	$[\![*]\!]_{\mathbf{A},g} = *_A$
Variables:	$[\![x]\!]_{\mathbf{A},g} = g(x)$
Terms:	$[\![t_1 * t_2]\!]_{\mathbf{A},g} = [\![t_1]\!]_{A,g} *_A [\![t_2]\!]_{\mathbf{A},g}$

For formulas $[\![\varphi]\!]_{A,g}$ specifies truth conditions:

Atomic formulas:	$[\![R(t_1,t_2)]\!]_{\mathbf{A},g} = 1$ iff $\langle [\![t_1]\!]_{\mathbf{A},g}, [\![t_2]\!]_{\mathbf{A},g} \rangle \in R_A$; 0 otherwise
	$[\![t_1 = t_2]\!]_{\mathbf{A},g} = 1$ iff $[\![t_1]\!]_{\mathbf{A},g} = [\![t_2]\!]_{\mathbf{A},g}$; 0 otherwise
Formulas:	$[\![\neg\varphi]\!]_{\mathbf{A},g} = 1$ iff $[\![\varphi]\!]_{\mathbf{A},g}$ 0; 0 otherwise
	$[\![\varphi \wedge \psi]\!]_{\mathbf{A},g} = 1$ iff $[\![\varphi]\!]_{\mathbf{A},g} = 1$ and $[\![\psi]\!]_{\mathbf{A},g} = 1$; 0 otherwise

$[\![\varphi \vee \psi]\!]_{\mathbf{A},g} = 1$ iff $[\![\varphi]\!]_{\mathbf{A},g} = 1$ or $[\![\psi]\!]_{\mathbf{A},g} = 1$; 0 otherwise
$[\![\exists x\varphi]\!]_{\mathbf{A},g} = 1$ iff for some $a \in A$: $[\![\varphi]\!]_{\mathbf{A},gx^a} = 1$; 0 otherwise
$[\![\forall x\varphi]\!]_{\mathbf{A},g} = 1$ iff for every $a \in A$: $[\![\varphi]\!]_{\mathbf{A},gx^a} = 1$; 0 otherwise

Let **A** and **B** be structures for L and let $h: A \to B$ be *a homomorphism*.

For assignment g on A we define assignment h[g] on B as:
for all $x \in VAR$: $h[g](x) = h(g(x))$

Lemma: If $t \in TERM_L$ and h is a homorphism then $h([\![t]\!]_{A,g}) = [\![t]\!]_{A,h[g]}$

Proof: with induction
1. $h([\![s]\!]_{\mathbf{A},g}) = h(s_A) = s_B = [\![s]\!]_{\mathbf{B},h[g]}$
2. $h([\![x]\!]_{\mathbf{A},g}) = h(g(x)) = h[g](x) = [\![x]\!]_{\mathbf{B},h[g]}$
3. Assume that $h([\![t_1]\!]_{\mathbf{A},g}) = [\![t_1]\!]_{\mathbf{B},h[g]}$ and
$h([\![t_2]\!]_{\mathbf{A},g}) = [\![t_2]\!]_{\mathbf{B},h[g]}$
Then: $h([\![t_1 * t_2]\!]_{\mathbf{A},g}) = h([\![t_1]\!]_{\mathbf{A},g} *_{\mathbf{A}} [\![t_2]\!]_{\mathbf{A},g}) = h([\![t_1]\!]_{\mathbf{A},g}) *_B h([\![t_2]\!]_{\mathbf{A},g})$ [homomorphism]
$h([\![t_1]\!]_{\mathbf{A},g}) *_{\mathbf{B}} h([\![t_2]\!]_{\mathbf{A},g}) = [\![t_1]\!]_{\mathbf{B},h[g]} *_{\mathbf{B}} [\![t_2]\!]_{\mathbf{B},h[g]}$ [induction assumption]
$[\![t_1]\!]_{\mathbf{B},h[g]} *_{\mathbf{B}} [\![t_2]\!]_{\mathbf{B},h[g]} = [\![t_1 * t_2]\!]_{\mathbf{B},h[g]}$
In sum: $h([\![t_1 * t_2]\!]_{\mathbf{A},g}) = [\![t_1 * t_2]\!]_{\mathbf{B},h[g]}$
□

Theorem: If φ is a positive formula and h is a surjective homomorphism then
If $[\![\varphi]\!]_{A,g} = 1$ then $[\![\varphi]\!]_{\mathbf{B},h[g]} = 1$
Positive formulas are preserved onto homomorphic images

Proof: with induction
1. Assume $[\![R(t_1,t_2)]\!]_{\mathbf{A},g} = 1$
Then $<[\![t_1]\!]_{A,g}, [\![t_2]\!]_{\mathbf{A},g}> \in R_{\mathbf{A}}$
Then $<h([\![t_1]\!]_{\mathbf{A},g}), h([\![t_2]\!]_{\mathbf{A},g})> \in R_{\mathbf{B}}$ [homomorphism]
Then $<[\![t_1]\!]_{\mathbf{B},h[g]}), [\![t_2]\!]_{\mathbf{B},h[g]})> \in R_{\mathbf{B}}$ [lemma]
Then $[\![R(t_1,t_2)]\!]_{\mathbf{B},h[g]} = 1$

2. Assume $[\![(t_1 = t_2)]\!]_{\mathbf{A},g} = 1$
Then $[\![t_1]\!]_{\mathbf{A},g} = [\![t_2]\!]_{\mathbf{A},g}$
Then $h([\![t_1]\!]_{\mathbf{A},g}) = h([\![t_2]\!]_{\mathbf{A},g})>$ [homomorphism]
Then $[\![t_1]\!]_{\mathbf{B},h[g]}) = [\![t_2]\!]_{\mathbf{B},h[g]})$ [lemma]
Then $[\![(t_1 = t_2)]\!]_{\mathbf{B},h[g]} = 1$

3. Assume: If $[\![\varphi]\!]_{\mathbf{A},g} = 1$ then $[\![\varphi]\!]_{\mathbf{B},h[g]} = 1$ and
If $[\![\psi]\!]_{\mathbf{A},g} = 1$ then $[\![\psi]\!]_{\mathbf{B},h[g]} = 1$
Assume $[\![\varphi \wedge \psi]\!]_{\mathbf{A},g} = 1$.
Then $[\![\varphi]\!]_{\mathbf{A},g} = 1$ and $[\![\psi]\!]_{\mathbf{A},g} = 1$
Hence $[\![\varphi]\!]_{\mathbf{B},h[g]} = 1$ and $[\![\psi]\!]_{\mathbf{B},h[g]} = 1$ [induction assumption]
Hence $[\![\varphi \wedge \psi]\!]_{\mathbf{B},h[g]} = 1$

4. Assume: If $[\![\varphi]\!]_{\mathbf{A},g} = 1$ then $[\![\varphi]\!]_{\mathbf{B},h[g]} = 1$ and
If $[\![\psi]\!]_{\mathbf{A},g} = 1$ then $[\![\psi]\!]_{\mathbf{B},h[g]} = 1$
Assume $[\![\varphi \vee \psi]\!]_{\mathbf{A},g} = 1$.
Then $[\![\varphi]\!]_{\mathbf{A},g} = 1$ or $[\![\psi]\!]_{\mathbf{A},g} = 1$
Hence $[\![\varphi]\!]_{\mathbf{B},h[g]} = 1$ or $[\![\psi]\!]_{\mathbf{B},h[g]} = 1$ [induction assumption]
Hence $[\![\varphi \vee \psi]\!]_{\mathbf{B},h[g]} = 1$

5. Assume for every $a \in A$: if $[\![\varphi]\!]_{\mathbf{A},g_x{}^a} = 1$ then $[\![\varphi]\!]_{\mathbf{B},h[g]_z{}^{h(a)}} = 1$
Assume $[\![\exists x\varphi]\!]_{\mathbf{A},g} = 1$
Then for some $a \in A$: $[\![\varphi]\!]_{\mathbf{A},g_x{}^a} = 1$
Then for some $a \in A$: $[\![\varphi]\!]_{\mathbf{B},h[g]_x{}^{h(a)}} = 1$ [induction assumption]
Then for some $b \in B$: $[\![\varphi]\!]_{\mathbf{B},h[g]_x{}^b} = 1$ [namely: $b = h(a)$]
Then $[\![\exists x\varphi]\!]_{\mathbf{B},h[g]} = 1$

6. Assume for every $a \in A$: if $[\![\varphi]\!]_{\mathbf{A},g_x{}^a} = 1$ then $[\![\varphi]\!]_{\mathbf{B},h[g]_z{}^{h(a)}} = 1$
Assume $[\![\forall x\varphi]\!]_{\mathbf{A},g} = 1$
Then for every $a \in A$: $[\![\varphi]\!]_{\mathbf{A},g_x{}^a} = 1$
Then for every $a \in A$: $[\![\varphi]\!]_{\mathbf{B},h[g]_x{}^{h(a)}} = 1$ [induction assumption]
Then for every $b \in B$: $[\![\varphi]\!]_{\mathbf{B},h[g]_x{}^b} = 1$ because h is a surjection: every b is h(a) for some $a \in A$]
Then $[\![\forall x\varphi]\!]_{\mathbf{B},h[g]} = 1$

Consequences:
Example 1:
Let <A,R> be a connected reflexive structure:
$\forall x[R(x,x)]$
$\forall x \forall y[R(x,y) \vee R(y,x) \vee x=y]$
Let h: <A,R> $\rightarrow$ <B,R> be a surjective homomorphism

Then <B,R> is also connected reflexive structure.

Example 2:
Let <A,*> be an itempotent, commutative, associative structure:
$\forall x.[x*x = x]$
$\forall x \forall y[x*y = y*x]$
$\forall x \forall y \forall z[\ x*(y*z) = (x*y)*z]$
Let h: <A,*> $\rightarrow$ <B,*> be a surjective homomorphism

Then <B,*> is a;lso an idempotent, commutative, associative structure.

The next notion is an algebraic notion.

Let **A** = <A, R, *, s_1, s_2> be a structure.

A **congruence relation** on **A** is an equivalence relation $\approx_A$ that preserves the algebraic structure:

1. for all $a_1,a_2,b_1,b_2 \in A$:
   if $a_1 \approx_A a_2$ and $b_1 \approx_A b_2$ then $(a_1 * b_1) \approx_A (a_2 * b_2)$
2. $\neg(s_1 \approx_A s_2)$

Congruence relations (like homomorphisms) preserve the algebraic structure (the special elements are not congruent, just like homomorphisms do not identify special elements).

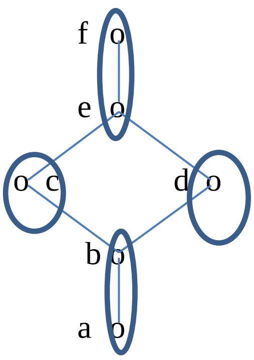
Let **A** = <A, R, *, s> be a structure and let $\approx_A$ be a congruence relation on **A**.
We define the **congruence structure of A under ≈: $A_\approx$**

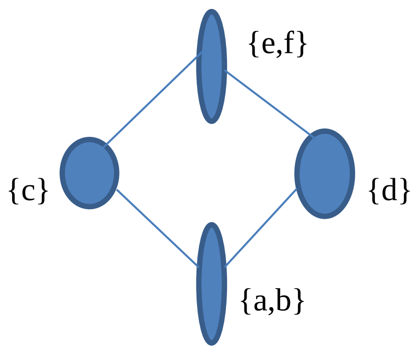
$\mathbf{A}_\approx = \langle A_\approx, R_\approx, *_\approx, s_\approx \rangle$ where

1. $A_\approx = [A]_\approx$
2. for all $a,b \in A$:
   $R_\approx([a]_\approx,[b]_\approx)$ iff $\exists x \in [a]_\approx \exists y \in [b]_\approx$: $R(x,y)$
3. for all $a,b \in A$: $([a]_\approx *_\approx [b]_\approx) = [a * b]_\approx$
4. $s_\approx = [s]_\approx$

Example.

A = <{a,…,f}, ⊑,⊔>





⊑ is a partial order
⊔ is join

≈: as in the picture e≈f, a≈b,…

Preservation of ⊑:
{a,b} $\sqsubseteq_\approx$ {c} because a ⊑ c
{c} $\sqsubseteq_\approx$ [e,f} because c ⊑ f etc…

Preservation of ⊔:
$\{c\} \sqcup_\approx \{f\} = [c]_\approx \sqcup_\approx [d]_\approx = [c \sqcup d]_\approx = [e]_\approx = \{e,f\}$

Substructures:
A↾{b,c,d,e} is a substructure of **A**
A↾{a,c,d,f} is not a substructure of **A**

Let **A** be a structure, let $\approx$ be a congruence relation on **A**, and let $\mathbf{A}_\approx$ be the congruence structure of **A** under $\approx$.

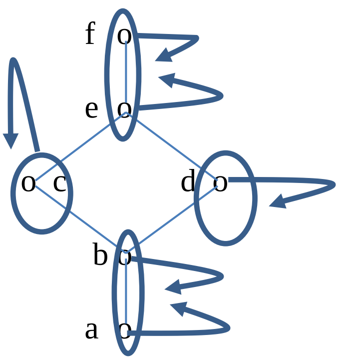
The **natural homomorphism on A relative to** $\approx$, $h_\approx$ is the function $h_\approx$ from A onto $A_\approx$ such that:

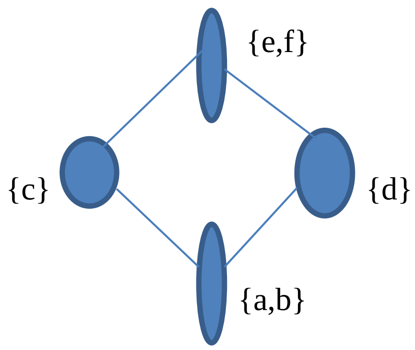
for all $a \in A$: $h_\approx(a) = [a]_\approx$

Lemma 1: The natural homomorphism $h_\approx$ is a surjective homomorphism from A onto $A_\approx$
Proof.
1. $h_\approx$ is onto. If $B \in A_\approx$ then for some $a \in A$: $B = [a]_\approx$. $h_\approx(a)=B$.
2. Let $a,b \in A$ and $R(a,b)$. Then $a \in [a]_\approx$ and $b \in [b]_\approx$ and $R(a,b)$, hence $R_\approx([a]_\approx,[b]_\approx)$. So indeed $R_\approx(h_\approx(a),h_\approx(b))$
3. $h_\approx(a * b) = [a * b]_\approx =$ (by the fact that $\approx$ is a congruence relation) $[a]_\approx *_\approx [b]_\approx = h_\approx(a) *_\approx h_\approx(b)$
4. $h_\approx(s) = [s]_\approx = s_\approx$

□

Example.
A = <{a,…,f}, ⊑,⊔>





$\approx$: as in the picture $e\approx f$, $a\approx b$,…
$h_\approx(e)=h_\approx(f)=\{e,f\}$
$h_\approx(a)=h_\approx(b)=\{a,b\}$
$h_\approx(c)=\{c\}$, $h_\approx(d)=\{d\}$

Substructures:
A↾{b,c,d,e} is a substructure of **A**
A↾{a,c,d,f} is not a substructure of **A**
(while the relation ⊑ is preserved, the operation ⊔ is not, since $c \sqcup d \notin \{a,c,d,f\}$.

Preservation of ⊑:
$\{a,b\} \sqsubseteq_\approx \{c\}$ because $a \sqsubseteq c$
$\{c\} \sqsubseteq_\approx [e,f\}$ because $c \sqsubseteq f$ etc…

Preservation of ⊔:
$h_{\approx}(c \sqcup d) = h_{\approx}(e) = [e]_{\approx} = \{e,f\}$
$h_{\approx}(c) \sqcup_{\approx} h_{\approx}(d) = \{c\} \sqcup_{\approx} \{d\}$ = [by definition] $[c \sqcup d]_{\approx} = [e]_{\approx} = \{e,f\}$ etc.
A congruence relation preserves the algebraic structure by **contracting elements**, identifying elements within the structure.

Lemma 2: If h is a homomorphism from **A** into **B** then h(**A**) is a substructure of **B**
Proof.
This fact was mentioned above: h(**A**) = **B**↾h(A).
-Obviously $h(A) \subseteq B$.
-h(A) is closed under $*_B$: If $b_1, b_2 \in h(A)$, then for some $a_1,a_2 \in A$: $b_1 = h(a_1)$ and $b_2 = h(a_2)$.
Then $b_1 *_B b_2 = h(a_1) *_B h(a_2)$ = (homomorphism) $h(a_1 * a_2)$, and indeed $b_1 *_B b_2 \in h(A)$.
-The restriction of R and s are unproblematic.
□

Lemma 3: If $\approx_A$ a congruence relation on **A** and and **C** a substructure of **A**, then:
$\approx_A$↾C is a congruence relation on **C**
Proof.
Let us set $\approx_C = \approx_A$↾C
-It is easy to see that $\approx_C$ is an equivalence relation on C.
-Obviously, since the special elements are in different equivalence classes in **A,** they stay in different equivalence classes in C, and since **C** is a substructure of **A**, C contains the special elements.
-let $a_1,a_2,b_1,b_2 \in C$ and let $a_1 \approx_C b_1$ and $a_2 \approx_C b_2$.
Then $(a_1 *_C a_2) \approx_A (b_1 *_C b_2)$
Since **C** is a substructure of **A**, $(a_1 *_C a_2) \in C$ and $(b_1 *_C b_2) \in C$.
Hence $(a_1 *_C a_2) \approx_C (b_1 *_C b_2)$
□

Lemma 4: If h is a homomorphism from **A** into **B** and **C** a substructure of **A**, then:
h↾C is a homomorphism from **C** into **B**
Proof.
-The preservation of the special elements follows from the fact that a substructure of **A** has the same special elements as **A**.
-The preservation of the relations and operations is obvious.
□

Lemma 5: The composition of two homomorphisms is a homomorphism
Proof.
Let f: **A** → **B** and g: **B** → **C** be homomorphisms.
-g ∘ f is a function from A into C.
-Let $R_A(a_1,a_2)$. Then, because f is a homomorphism, $R_B(f(a_1),f(a_2))$, with $f(a_1),f(a_2) \in B$.
Then, because g is a homomorphism, $R_C(g(f(a_1)),g(f(a_2)))$.
Hence $R_C(g\circ f(a_1), g\circ f(a_2))$
- for every $a_1,a_2 \in A$: $g\circ f(a_1) *_C g\circ f(a_2) = g(f(a_1)) *_C g(f(a_2)) = g((f(a_1) *_B f(a_2)))$ (because g is a homomorphism) $= g(f((a_1 *_A a_2))$ (because f is a homomorphism) $= g \circ f(a_1 *_A a_2)$
-$g \circ f(s_A) = g(f(s_A)) = g(s_B) = s_C$.
□

Homomorphism Theorem.
Let h: A $\rightarrow$ B be a homomorphism and define $\approx_h$ and $f_{\approx,h}$ as follows:

for all $a_1, a_2 \in A$: $a_1 \approx_h a_2$ iff $h(a_1)=h(a_2)$
for all $a \in A$: $f([a]_{\approx h}) = h(a)$

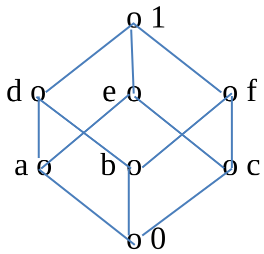
Then $\approx_h$ is a congruence relation on A and
f is an isomorphism between $A_{\approx h}$ and h(A)

Proof.
1. $\approx_h$ is a congruence relation on A.
It is easy to check that $\approx_h$ is an equivalence relation (it is built from identity).
Let $a_1 \approx_h b_1$ and $a_2 \approx_h b_2$, i.e. $h(a_1)=h(b_1)$ and $h(a_2)=h(b_2)$, then $h(a_1) *_B h(a_2) = h(b_1) *_B h(b_2)$, and hence (by the fact that h is a homomorphism)
$h(a_1 *_A a_2) = h(b_1 *_A b_2)$, and hence $a_1 *_A a_2 \approx_h b_1 *_A b_2$
So $\approx_h$ is a congruence relation.

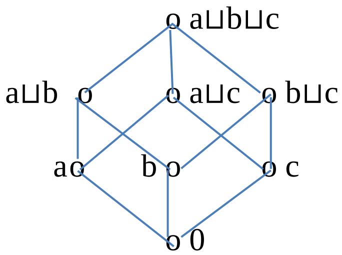
2.- $f([a_1]_{\approx h} *_{\approx h} [a_2]_{\approx h}) = f([a_1 *_A a_2]_{\approx h}) = h(a_1 *_A a_2) =$ (homomorphism)
$h(a_1) *_B h(a_2) = f([a_1]_{\approx h}) *_B f([a_2]_{\approx h})$
Again, special elements are obviously preserved, hence f is a homomorphism.
Let $b \in h(A)$. Then for some $a \in A$: $h(a) \in B$. Then $f([a]_{\approx h})=b$, so f is onto.
Assume $h(a_1) = h(a_2)$. Then $a_1 \approx_h a_2$ and $[a_1]_{\approx h} = [a_2]_{\approx h}$. So f is one-one.
□

SOME BOOLEAN ILLUSTRATIONS

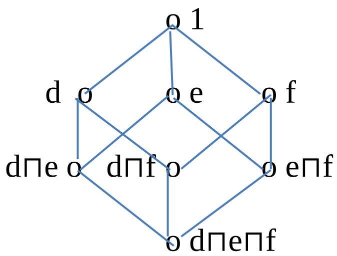
We introduce Boolean algebras property in chapter 3. But it will be already useful to use these richer structures to illustrate the restrictions on homomorphisms and congruence relations. For us in this section a Boolean algebra is a structure **B** = <B,⊑,¬,⊓,⊔,0,1> which has one special relation, partial order ⊑, one one-place operation ¬ of complementation, two two-place operations ⊓ and ⊔ of join and meet and two special elements 0 and 1.
We illustrated the notions of 0, 1 and join and meet in pictures above (repeated here):



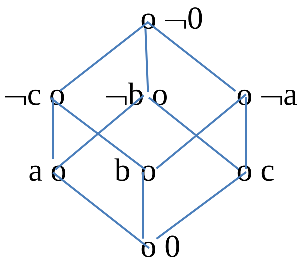
0 and 1 are resp. the minimal and maximal element of B.

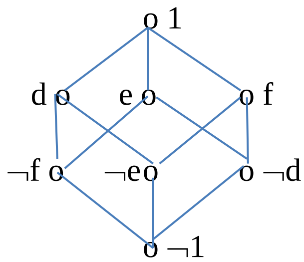


a⊔b, the join of a and b, is the smallest element of B such that a ⊑ a⊔b and b ⊑ a⊔b



d⊓e, the meet of d and e, is the largest element of B such that d⊓e ⊑ a and d⊓e ⊑ f





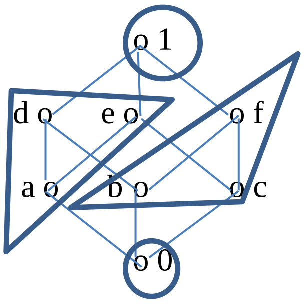
We say: x and y do not overlap iff x⊓y=0.
x and y do not overlap if the only part they have in common is 0 (they have no 'real' part in common). With this we define:

¬x, the complement of x, is the maximal element of B that does not overlap x.

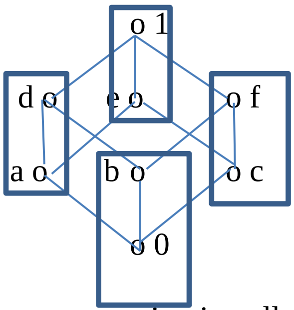
In order to be Boolean algebras, structures need to satisfy special postulates which we ignore here: we know enough to talk about congruence relations and homomorphisms.

First we illustrate the difference between a relation that is only an equivalence relation and a relation that is an congruence relation:

B_1



B_2



no preservation in collapsing elements   preservation in collapsing elements

B_1 is an equivalence relation, but not a congruence relation:

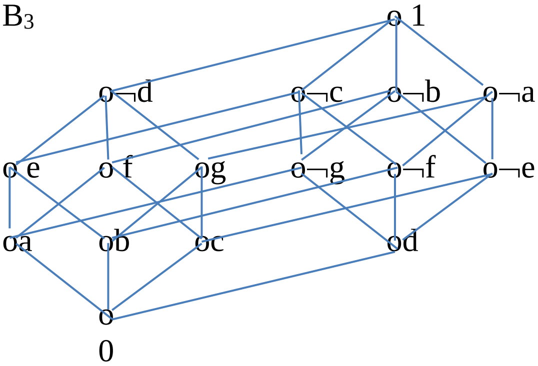
a ≈ a and b ≈ f. If the relation were a congruence relation, what should hold is that: a⊔b ≈ a⊔f. But a⊔b = d and a⊔f = 1, and ¬(d ≈ 1).

B_2 is both an equivalence relation and a congruence relation.
For instance, a≈d and c≈f, and a⊔c = e and d⊔f=1, and indeed e≈1.

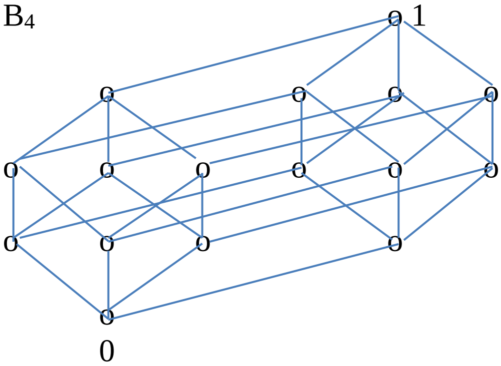
The intuitive difference is visible: both equivalence relations contract the 8-element structure to a 4-element structure, but the second one does so by collapsing what was a *plane* in the 8-element structure onto another *plane*: that is, you collapse something that has *itself* the structure of a Boolean algebra (but *not* a sub-Boolean algebra) onto a non-overlapping structure which *also itself* has the structure of a Boolean algebra. This is what congruences do.

We are now interested in homomorphisms from B_3 to B_4, in particular in a homomorphism that contracts the structure B_3 to a structure of 8 elements.

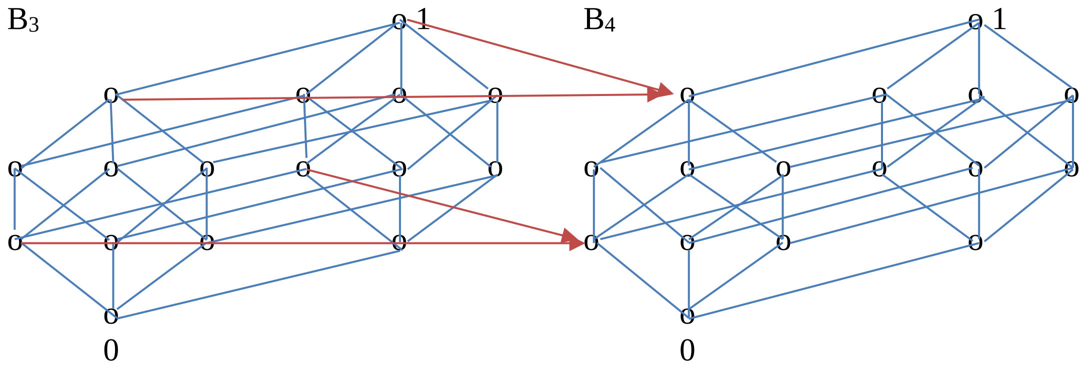
B_3



B_4

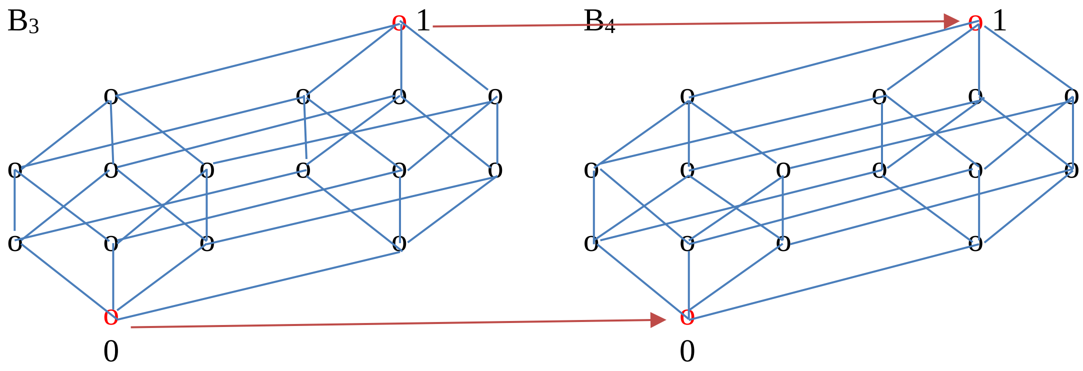


If the structure had been defined as a structure <B,⊑,⊓,⊔>, without mention of special elements or operation ¬, then forming a homomorphism would be simple, because you see the 8-element structures sitting in the picture, so, the following would do, with the other arrows analogously identifying corresponding elements.

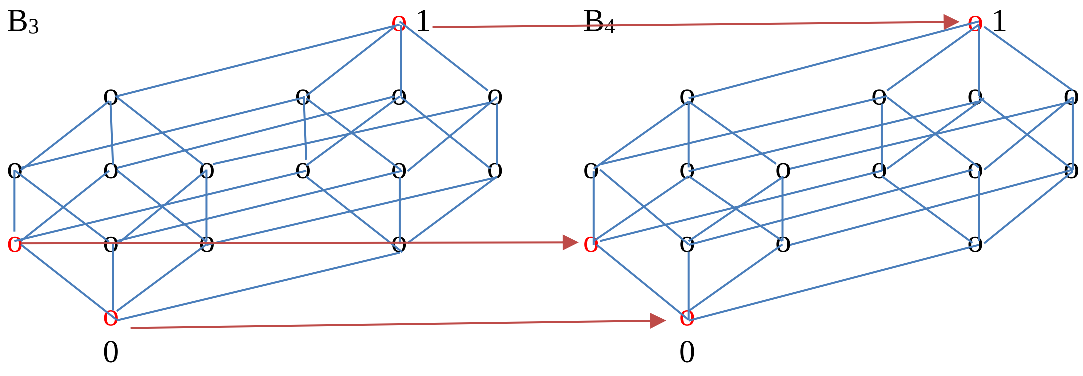


But this function is not a Boolean homomorphism, because complements and special elements are not preserved.

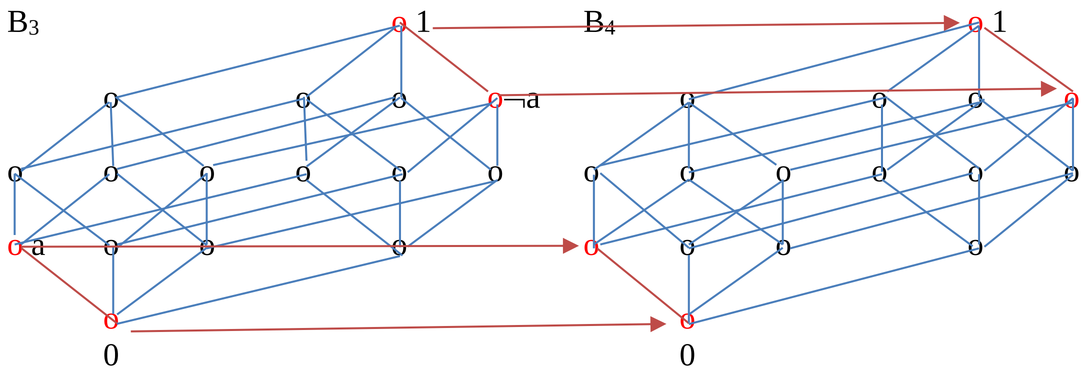
Thus, a first fact about preservation is that only functions that preserve the special elements 0 and 1 will be able to count as homomorphisms:



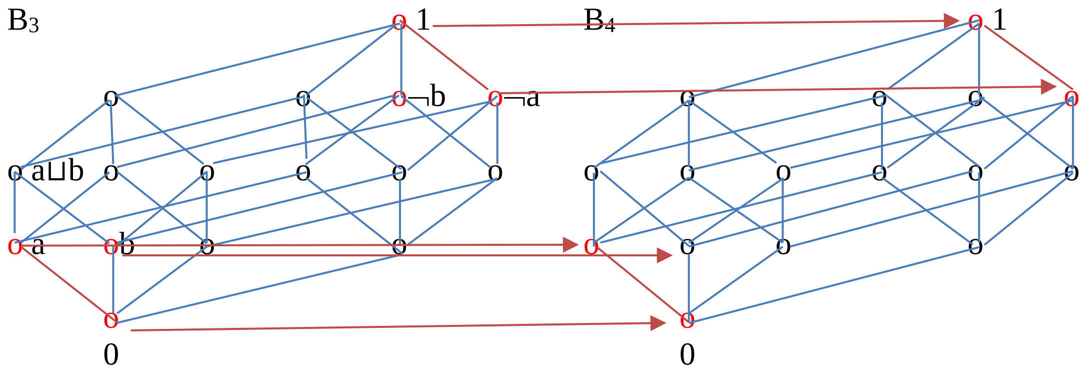
Let us add one more connection to the homomorphism:



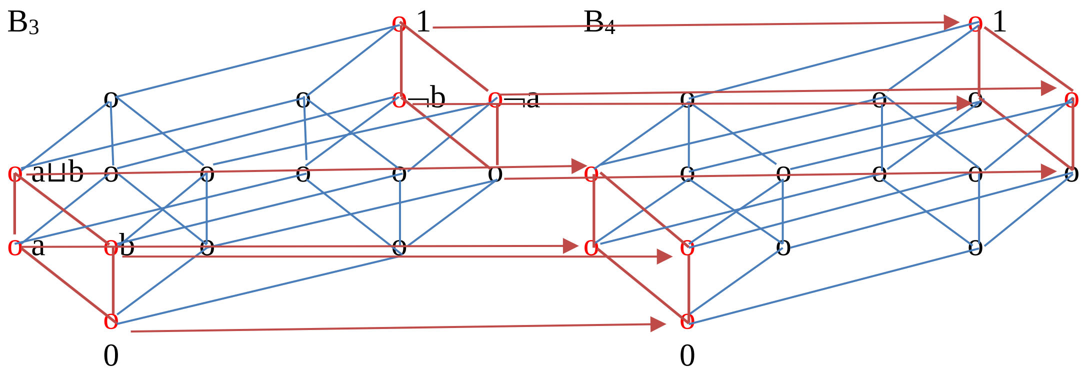
The fact that the homorphism is a Boolean homomorphism, says that complements must be preserved: this means that ¬f(a) must be mapped onto f(¬a):



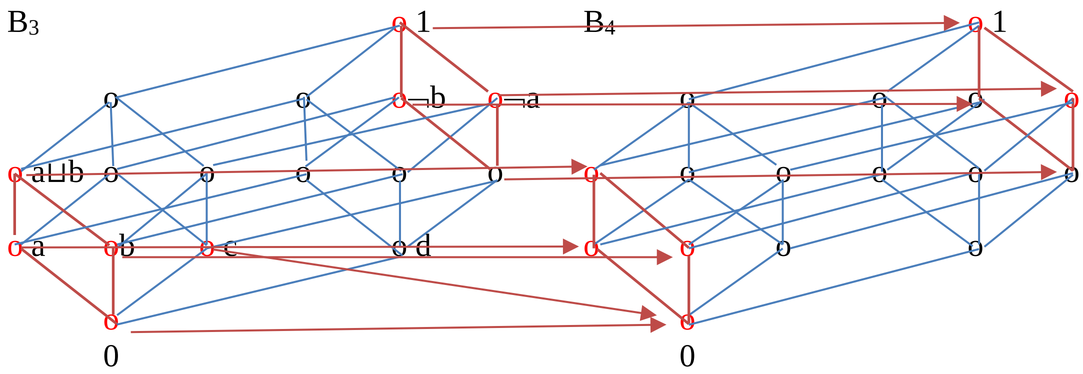
Next, we decide that b will be mapped independently:



This fixes the values for ¬b and for a⊔b:

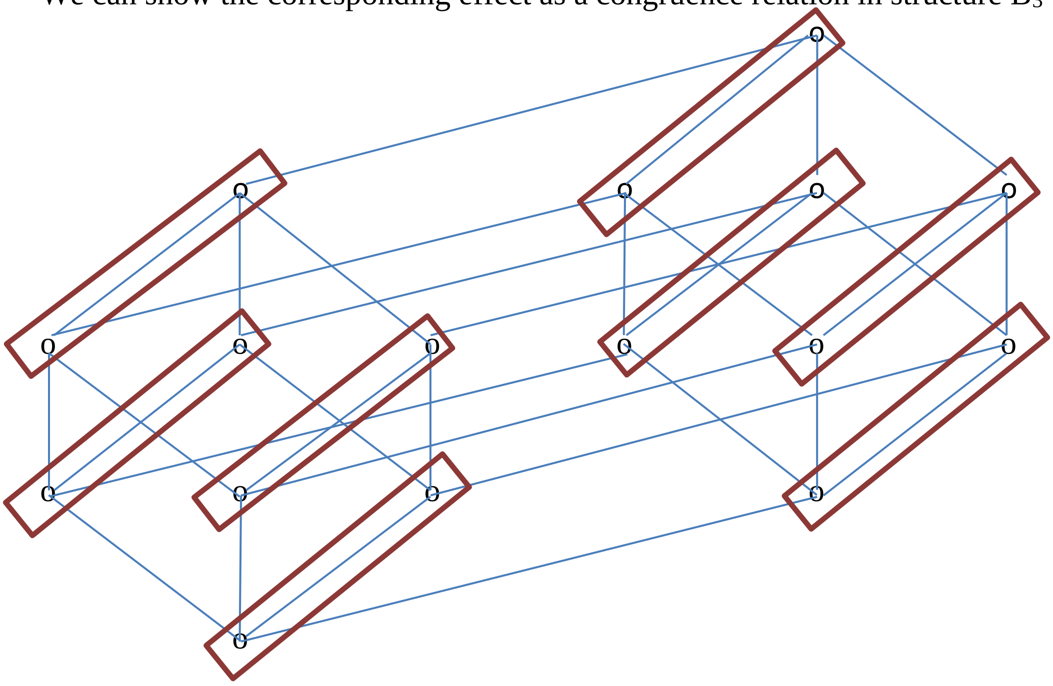


We have fixed 8 elements in the value structure. This means that we will have to identify all the remaining elements with the ones already determined (since we wanted 8 elements). Fixing one more element will fix the whole structure. The intuition of collapsing planes tells you that you have the choice of collapsong either c or d with 0. We choose the visually easiest and map c onto 0:

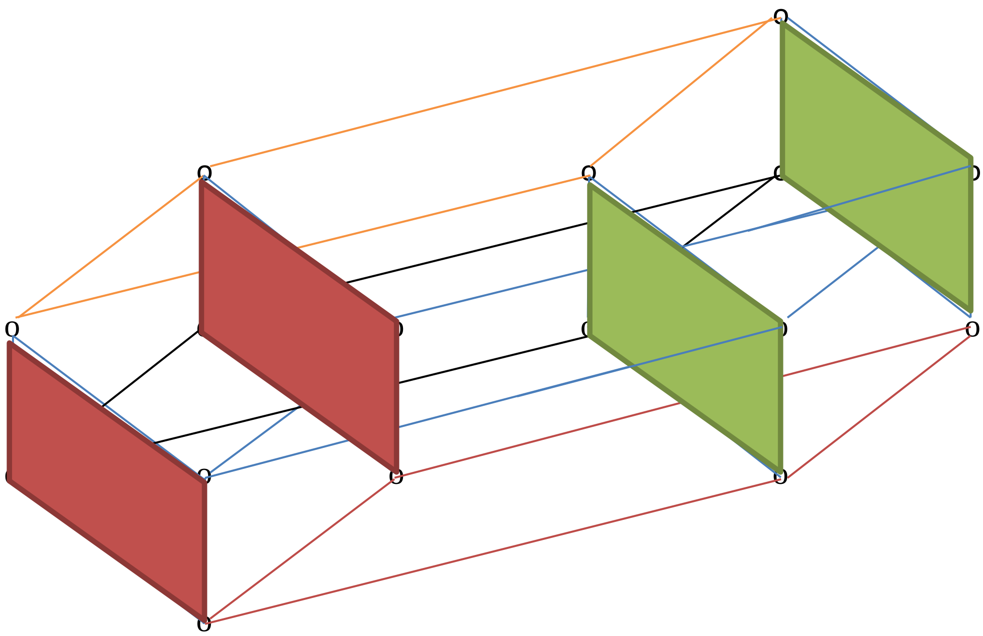


The effect is that the remaining left blue plane is collapsed onto the left red plane, and the remaining right blue plane onto the remaining right red plane.

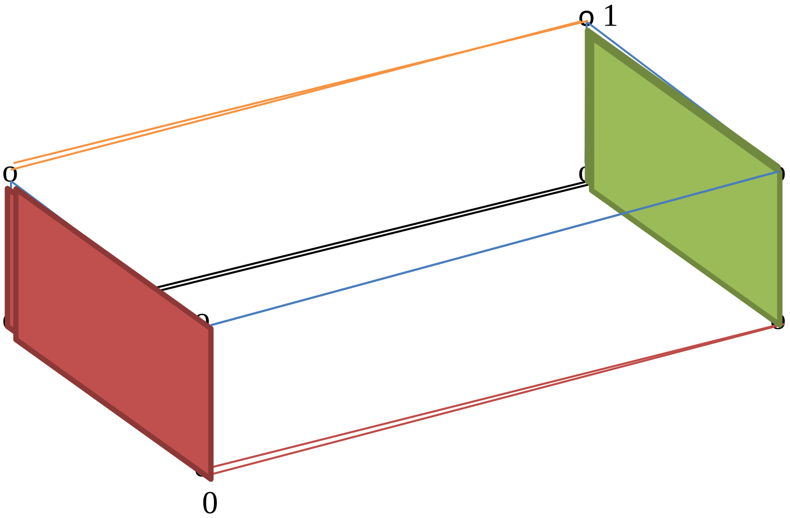
We can show the corresponding effect as a congruence relation in structure B_3



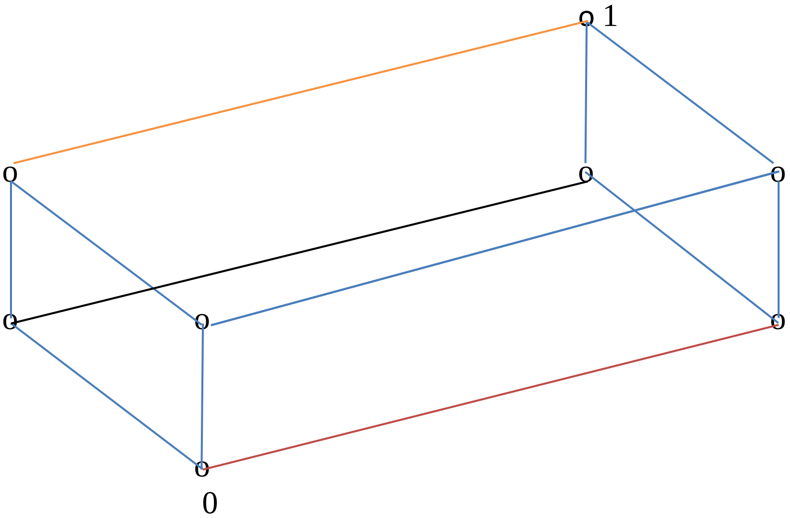
The effect is that the planes are collapsed as follows:



In this process, the horizontal planes get collapsed into lines:

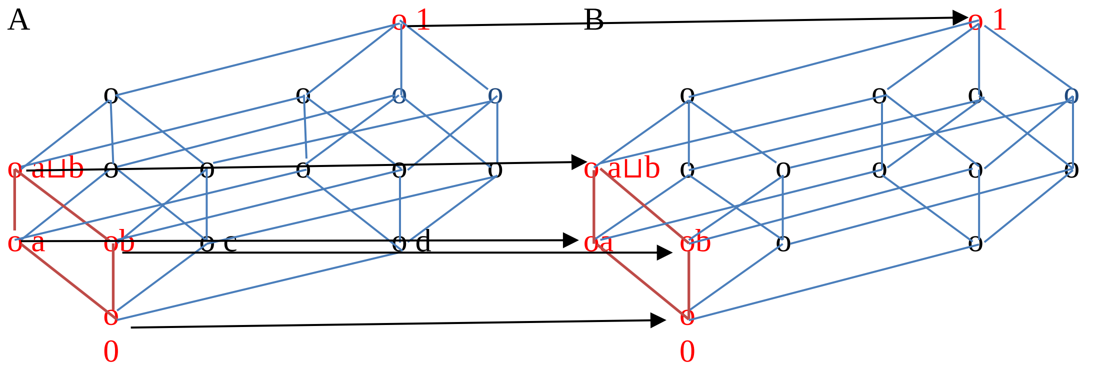


And, of course, this gives an 8-element Boolean algebra

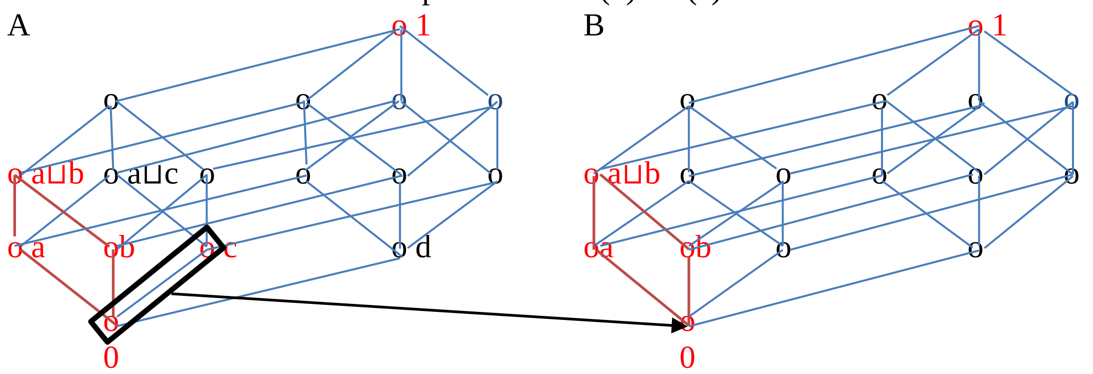


Once more the story:

We want a homomorphism that maps 16 element Boolean algebra A onto an 8 element Boolean algebra in B (which, for ease we take to be A).
-We start out by mapping 0 onto 0 and 1 onto 1: h(0) = 0 and h(1) = 1
-We choose to map a onto a and b onto b, h(a) = a and h(b) = b
This means that a⊔b will be mapped onto a⊔b, because h(a⊔b) = h(a) ⊔ h(b) = a⊔b:



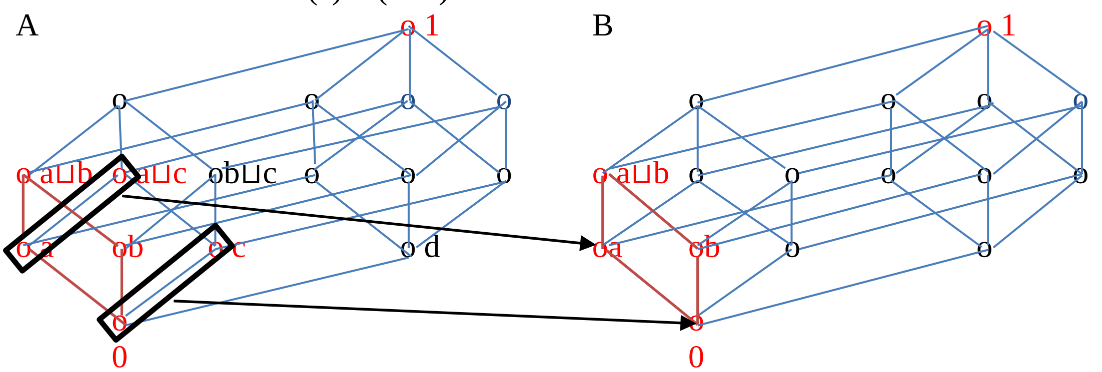
Now we decide to map c onto 0: h(c) = 0.
This means that both 0 and c map onto c: h(0) = h(c) = 0



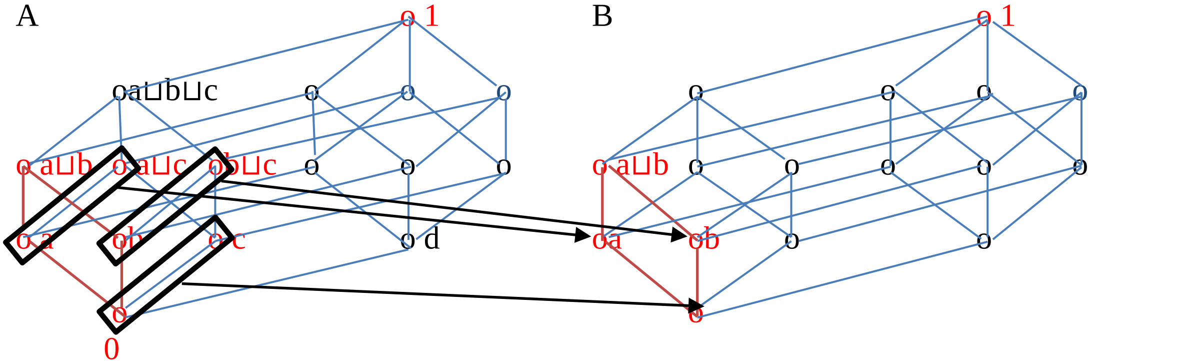
Now we argue:

(1) h(a) = a
(2) h(a)⊔0 = a [since a⊔0=a]
(3) h(c)=0
(4) h(a)⊔h(c) = a [by substituting h(c) for 0 in (2)]
(5) h(a⊔c) = a [from (4), since h is a homomorphism]

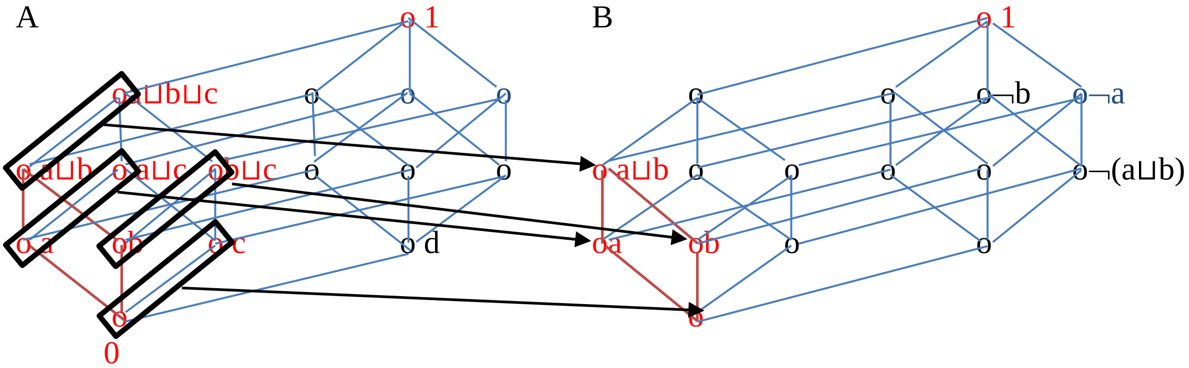
Hence it follows that h(a)=h(a⊔c):



Exactly the same argument shows that h(b⊔c)=h(b)=b:

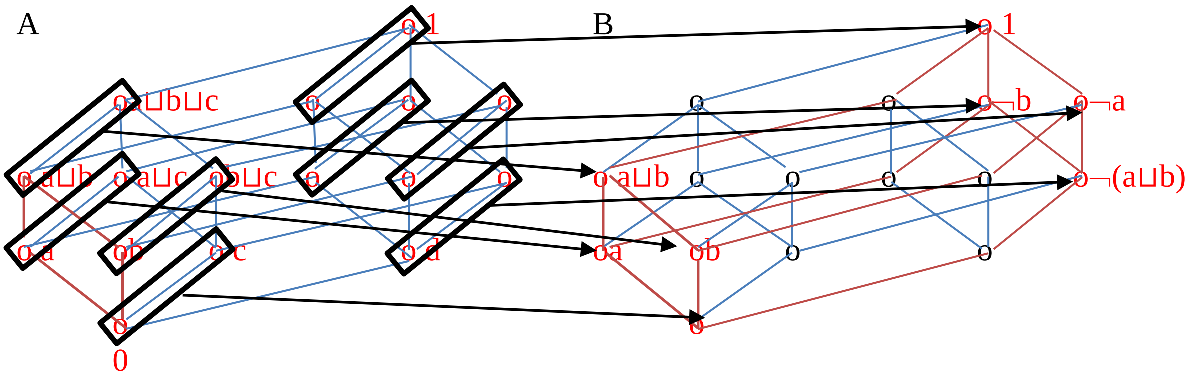


And with h(a⊔b) = a⊔b and h(c)=0, of course h(a⊔b⊔c) = a⊔b:¬



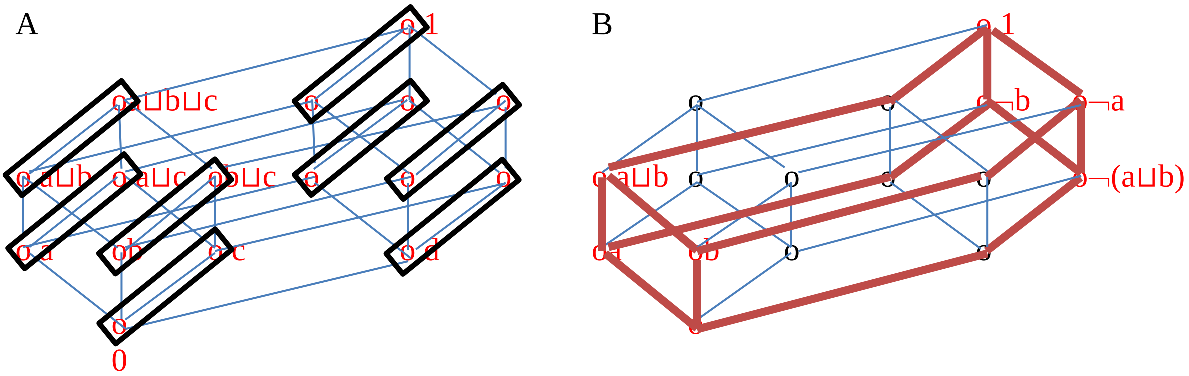
So the homomorphism contracts the elements in the blocks to the lowest element.

With duality, the same argument will show that the homomorphism divides the remaining elements similarly into blocks and contracts the elements in these blocks to the higher element:
(i.e. ¬c is mapped onto 1).



Now we see clearly the relations between homomorphisms, congruence relations and substructures:

homomorphism h:**A** → **B** determines a ***congruence relation*** $\sim_h$ on **A** and its range is a ***substructure*** of **B**:



Thus: on algebras homomorphisms and congruence relations are two sides of the same coin.

Equational classes of algebras.

One more general algebraic notion is that of product:

Let **A** = $<A, R_A, *_A, s_A>$ and **B** = $<B, R_B, *, s_A>$ be structures of the same type.
The **direct product of A and B**, **A×B**, is:

A×B = $<A\times B, R_{A\times B}, *_{A\times B}, s_{A\times B}>$, where:

1. $R_{A\times B}(<a_1,b_1>, <a_2,b_2>)$ iff $R_A(a_1,a_2)$ and $R_B(b_1,b_2)$
2. $(<a_1,b_1> *_{A\times B} <a_2,b_2>) = <(a_1 *_A a_2), (b_1 *_A b_2)>$
3. $s_{A\times B} = <s_A,s_{B>}$

K is an **equational class of algebras** iff *K* is a class of algebras of the same type and:

1. If **A** ∈ *K* and **B** is a subalgebra of **A** then **B** ∈ *K*
2. If **A** ∈ *K* and h is a homorphism from **A** onto **B** then **B** ∈ *K*
3. If **A** ∈ *K* and **B** ∈ *K* then the direct product **A**×**B** ∈ K

An **equational class of algebras** is a class of algebras of the same type which is closed under the formation of subalgebras, homomorphic images and direct products.

I repeat the definition of positive formulas:

Assume a first-order logical language $L_{*,s}$ for algebra with two-place operator-symbol * and individual constant s (again, algebra: we ignore relations here).

In the language $L_{*,s}$ all atomic formulas are of the form $(\alpha = \beta)$, where $\alpha,\beta \in TERM_{L*s}$ and $TERM_{:*s}$ is defined as:

$TERM_{L*s}$ is the smallest set such that:

1. VAR $\subseteq$ $TERM_{L*s}$ (all variables are terms)
2. s $\in$ $TERM_{L*s}$
3. if $t_1,t_3 \in TERM_{L*s}$ then $(t_1 * t_2) \in TERM_{L*s}$

A **positive formula in $L_{*,s}$** is a formula built from atomic formulas without the use of negation.

Theorem: Positive formulas are preserved under subalgebras, onto homomorphic images and onto direct products.

This means the following:
Let **A** = <A, $*_A$, s_A> be an algebra and φ a formula of $L_{*,s}$.
φ is true on **A** iff for every interpretation function F_A: φ is true on model <**A**, F_A>.

If φ is a positive formula and φ is true on **A** and on **B** then:
-If **C** is a subalgebra of **A**, or **C** is the homomorphic image of **A** under some homomorphism h, or **C** is the direct product of **A** and **B**, then φ is true on **C**.

Equational classes of algebras are classes of algebras that are defined by axiom schemas which are positive formulas.
This is the rationale for defining algebraic structures via identities. Algebraic identities are positive formulas.

So, when we claim that an algebraic operation * is commutative, we express that as the identity:

$(a * b) = (b * a)$

This stands for the positive formula:

$\forall x \forall y((x * y) = (y * x))$

The class of algebras where this formula holds is the class of commutative algebras. Since the axiom is a positive formula, this class is an equational class: closed under subalgebras, homomorphc images and direct products.

Building structures via equivalence relations

Lemma: Let **A** = <A, R> be a pre-order (reflexive and transitive)
Let $\approx_R$ the relation such that for all a,b $\in$ A: a $\approx_R$ b iff R(a,b) and R(b,a)

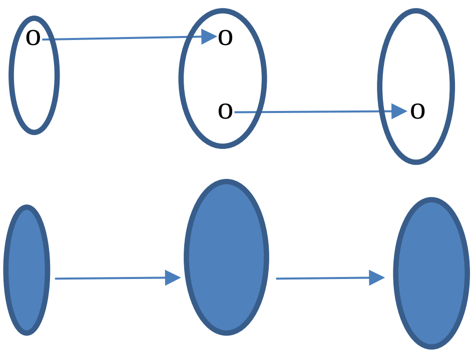
$\boldsymbol{\approx_R}$ is an equivalence relation and <$\mathbf{A}_{\approx R}$,$R_{\approx R}$> is a partial order.

Proof.
It is easy to check that $\approx_R$ is an equivalence relation.
-$R_{\approx R}$ is reflexive
for any a: a $\in$ $[a]_{\approx R}$ and R(a,a), hence $R_{\approx R}([a]_{\approx R},[a]_{\approx R})$
-$R_{\approx R}$ is transitive
Let $R_{\approx R}([a]_{\approx R},[b]_{\approx R})$ and $R_{\approx R}([b]_{\approx R},[c]_{\approx R})$
Say: x $\in$ $[a]_{\approx R}$ and y $\in$ $[b]_{\approx R}$ and R(x,y)
z $\in$ $[b]_{\approx R}$ and w $\in$ $[c]_{\approx R}$ and R(z,w)

y,z $\in$ $[b]_{\approx R}$, hence R(y,z) and R(z,y), by definition of $\approx_R$.
So R(x,y) and R(y,z) and R(z,w), and hence, since R is transitive, R(x,w).
So: x $\in$ $[a]_{\approx R}$ and w $\in$ $[c]_{\approx R}$ and R(x,w)
Hence $R_{\approx R}([a]_{\approx R},[c]_{\approx R})$

-$R_{\approx R}$ is antisymmetric.
Assume $R_{\approx R}([a]_{\approx R},[b]_{\approx R})$ and $R_{\approx R}([b]_{\approx R},[a]_{\approx R})$
Say: x $\in$ $[a]_{\approx R}$ and y $\in$ $[b]_{\approx R}$ and R(x,y)
z $\in$ $[b]_{\approx R}$ and w $\in$ $[a]_{\approx R}$ and R(z,w)
Then, by definition of $\approx_R$, R(y,z) and R(z,y), so R(x,y) and R(y,z).
Then, by transitivity, R(x,z).
Also, by definition of $\approx_R$, R(x,w) and R(w,x), so R(z,w) and R(w,x)
Hence, by transitivity, R(z,x)
So R(x,z) and R(z,x), and hence, by definition of $\approx_R$, then x $\approx_R$ z, and hence a $\approx_R$ b.
So $[a]_{\approx R} = [b]_{\approx R}$
□

In a picture:



Let **A** = <A,R > be a pre-order and $\approx_<$ as above.
A is **$\boldsymbol{\approx_R}$-connected** iff for every x,y $\in$ A: R(x,y) or R(y,x) or (x $\approx_R$ y)

Fact: If **A** = <A,R> is a $\approx_R$- connected pre-order, then $[\boldsymbol{A}]_{\approx R}$ is a linear order.

Refinement structures

Let A be a set.
Let Π_A be the set of all partitions on A.
The partition structure, $\mathbf{\Pi_A}$, is the structure:

$\mathbf{\Pi_A} = <\Pi_A, \sqsubseteq_\Pi, \sqcup_\Pi, \sqcap_\Pi, 0_\Pi, 1_\Pi >$ where:

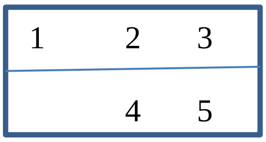
1. $P_A \sqsubseteq_\Pi Q_A$ iff $\forall B \in P_A\ \exists C \in Q_A: B \subseteq C$
2. $\sqcup_\Pi$ is defined below.
3. $P_A \sqcap_\Pi Q_A = \{B \cap C: B \in P_A \text{ and } C \in Q_A \text{ and } B \cap C \neq \emptyset\}$
4. $0_\Pi = \{\{a\}: a \in A\}$
5. $1_\Pi = \{A\}$

Facts: 1. $\sqsubseteq_A$ is a partial order
We call partial order $\sqsubseteq_\Pi$ the *refinement relation*: $P_A \sqsubseteq_\Pi Q_A$ means that partition P_A is a refinement of Q_A. In the diagrams this means, intuitively, that you get from Q_A to P_A by adding lines (i.e.by splitting blocks).

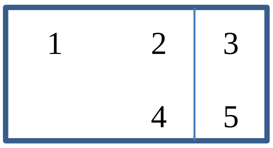
2. $P_A \sqcap_\Pi Q_A$ is the biggest partition of $\{P_A,Q_A\}$ such that $P_A \sqcap_\Pi Q_A \sqsubseteq_\Pi P_A$ and $P_A \sqcap_\Pi Q_A \sqsubseteq_\Pi Q_A$

$P_A \sqcap_\Pi Q_A$ is the minimal way of splitting blocks in P_A and in Q_A so that the result is a refinement both of P_A and of Q_A:

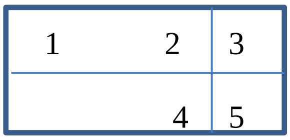
3. 0_Π is the minimum of Π_A, 0_Π is a refinement of every partition in Π_A
4. 1_Π is the maximum of Π_A, every partititio in Π_A is a refinement of 1_Π



P_A



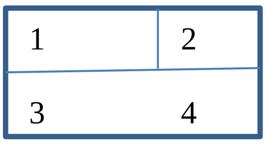
Q_A



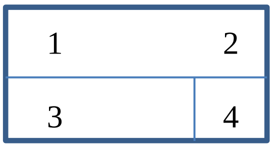
$P_A \sqcap_\Pi Q_A$

We come to $\sqcup_A$.
Fact: for every $P_A, Q_A \in \Pi_A$: there is a mimimal partition that both P_A and Q_A are refinements of, a mimimal way of unifying blocks in P_A and in Q_A into a partition that both P_A and Q_A are a refinement of.

We will make $P_A \sqcup_\Pi Q_A$ this partition:



P_A



Q_A

$P_A \sqcup_\Pi Q_A$

Intuitively, we remove all lines, except the ones that the two partitions share.
We follow the following steps:

Closure and +-closure under operation O.

Let $\langle A,O \rangle$ be a structure with $O: A^n \rightarrow A$ an n-place operation.
Let $B \subseteq A$.

$[B]^O$, *the closure of* B *under* O, is the smallest subset of A such that:

1. $B \subseteq [B]^O$
2. If $a_1,\ldots,a_n \in [B]^O$ then $O(a_1,\ldots,a_n) \in [B]^O$

Let $\langle A,O \rangle$ be a structure with $O: pow(A) \rightarrow A$ an operation.
Let $B \subseteq A$.

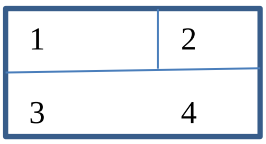
$[B]^O$, *the closuse of* B *under* O, is the smallest subset of A such that:

1. $B \subseteq [B]^O$
2. If $X \subseteq [B]^O$ then $O(X) \in [B]^O$

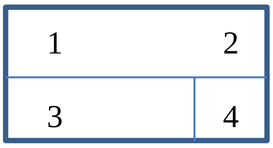
$[B]^{O+}$, *the +-closuse of* B *under* O, is the smallest subset of A such that:

1. $B \subseteq [B]^O$
2. If $X \subseteq [B]^O$ and $X \neq \emptyset$ then $O(X) \in [B]^O$

The join of the partitions on A: $P_A \sqcup Q_A$



$P_A = \{\{1\},\{2\},\{3,4\}\}$



$Q_A = \{\{1,2\},\{3\},\{4\}\}$

$P_A \sqcup Q_A = \{\{1,2\},\{3,4\}\}$

Intuitively, we remove all lines, except the ones that the two partitions share.

Formally:

Step 1: Take the +-closure of P_A and of Q_A:

$[P_A]^{\cup+} = \{\{1\},\{2\},\{3,4\}, \{1\}\cup\{2\},\{1\}\cup\{3,4\},\{2\}\cup\{3,4\},\{1\}\cup\{2\}\cup\{3,4\}\} =$
$\{\{1\},\{2\},\{3,4\}, \{1,2\}, \{1,3,4\}, \{2,3,4\}, \{1,2,3,4\}\}$

$[Q_A]^{\cup+} = \{\{1,2\},\{3\},\{4\},\{1,2\}\cup\{3\},\{1,2\}\cup\{4\},\{3\}\cup\{4\},\{1,2\}\cup\{3\}\cup\{4\}\} =$
$\{\{1,2\},\{3\},\{4\},\{1,2,3\}, \{1,2,4\}, \{3,4\} \{1,2,3,4\}\}$

Step 2: Take the intersection of $[P_A]^{\cup+}$ and $[Q_A]^{\cup+}$:

$[P_A]^{\cup+} \cap [Q_A]^{\cup+}$: $\{\{1,2\},\{3,4\},\{1,2,3,4\}\}$ This is a cover of A

min$_\sqsubseteq(A) = \{a \in A$: a is a minimal element of A under $\sqsubseteq\}$

Step 3: Take $\mathbf{min}_{\subseteq}([P_A]^{\cup+} \cap [Q_A]^{\cup+})$:

$\mathbf{min}_{\subseteq}([P_A]^{\cup+} \cap [Q_A]^{\cup+}) = \{\{1,2\},\{3,4\}\}$ This is a partition of A

Thus we define: $P_A \sqcup Q_A = \mathbf{min}_{\subseteq}([P_A]^{\cup+} \cap [Q_A]^{\cup+})$

Numbers

$\mathbb{N}$ is the set of natural numbers, {0,1,2,3,…}
$\mathbb{E}$ is the set of even numbers {0,2,4,6…}
$\mathbb{Z}$ is the set of integers {…−3,−2,−1,0,1,2,3 …}
$\mathbb{Z}^+$ is the set of positive integers {1,2,3,…}
$\mathbb{Q}$ is the set of rational numbers p/q with $p \in \mathbb{Z}$ and $q \in \mathbb{Z}^+$.
$\mathbb{R}$ is the set of real numbers p.X with $p \in \mathbb{Z}$ and X any countably infinite sequence of digits 0,1,2,3,4,5,6,7,8,9.

Cardinality

|A| is the cardinality of set A.
A and B have the same cardinality, |A| = |B| iff there is a bijection $h:A \rightarrow B$.

$\aleph_0$ is the cardinality of countably infinite sets: $|\mathbb{N}| = \aleph_0$
A is countably infinite iff $|A| = \aleph_0$

This means that A is countably infinite iff there is a bijection between $\mathbb{N}$ and A.

Cantor:
$|pow(A)| = 2^{|A|}$
Cantor's theorem: $2^{|A|} > |A|$, for any set A.
Proof: omitted

Fact 1: $\mathbb{N}$ is countably infinite.
By definition.

Fact 2: $\mathbb{E}$ is countably infinite.
The function $\lambda n.2n$ is a bijection between $\mathbb{N}$ and $\mathbb{E}$:

0	1	2	3	4	5	6	...
0	2	4	6	8	10	12	…

Fact 3: $\mathbb{Z}$ is countably infinite.
Make your bijection as follows: start in the middle at 0 and chose alternatingly the next positive and the next negative number:

0	1	2	3	4	5	6	7	8	...
0	1	−1	2	−2	3	−3	4	−4	…

Fact 4: $\mathbb{Q}$ is countably infinite

We show that the set of non-negative rational numbers is countably infinite.

Form a square in the following way:

0/1	0/2	0/3	0/4	0/5	0/6	…
1/1	1/2	1/3	1/4	1/5	1/6	…
2/1	2/2	2/3	2/4	2/5	2/6	…
3/1	3/2	3/3	3/4	3/5	3/6	…
4/1	4/2	4/3	4/4	4/5	4/6	…
5/1	5/2	5/3	5/4	5/5	5/6	…
6/1	6/2	6/3	6/4	6/5	6/6	…
...	...	...	...	...	...	

All non-negative rational numbers occur in this square.

Enumerate the square by following the diagonal:

0/1	0/2	0/3	0/4	0/5	0/6	…
1/1	1/2	1/3	1/4	1/5	1/6	…
2/1	2/2	2/3	2/4	2/5	2/6	…
3/1	3/2	3/3	3/4	3/5	3/6	…
4/1	4/2	4/3	4/4	4/5	4/6	…
5/1	5/2	5/3	5/4	5/5	5/6	…
6/1	6/2	6/3	6/4	6/5	6/6	…
...	...	...	...	...	...	

So we map 0 onto 0/1. We map the next natural number onto the next number on the diagonal that isn't yet in the mapping. So 1 is mapped onto 1/1, 2 is mapped onto 2/1; 3 is mapped onto 1/2; 4 is mapped onto 1/3, etc.

To get a bijection between ℕ and ℚ make a second square for the negative rational numbers and a diaginal enumeration, and alternate between adding the next number in the first square and adding the next number in the second square (i.e. do the same as what we did for ℤ).

Fact 5: |ℝ| = $2^{\aleph_0}$
Hence |ℝ| > |ℕ|

Fact 5 follows from the way the real numbers are constructed via countably infinite sets of rational numbers. The entailment that $|\mathbb{R}| > |\mathbb{N}|$ follows with Cantor's general theorem. There is an insightful direct proof of the latter, called a *diagonal proof*, which uses the representation of real numbers as consisting of an integer followed by a countably infinite sequence of digits.

We prove that the set of real numbers between 0 and 1, (0,1), is not countably infinite.

Assume (0,1) is countably infinite.
Then there is a countable list: $r = r_1, r_2, r_3\ldots$ which lists ***all*** the elements of (0,1).
We put them in the list vertically, horizontally we line up the digits in each countable sequence of digits. In this, q_1 is the postion of the first digit, q_2 the second, etc.

So $<r_n,q_m>$ is the digit that occurs on digit-place m in number r_n.
Hence $<r_2,q_{20}> = 7$ means hat the 20ieth digit in the digit list of number r_2 is 7

		q_1	q_2	q_3	q_4	…	q_n	…
r_1	0.	$<r_1,q_1>$	$<r_1,q_2>$	$<r_1,q_3>$	$<r_1,q_4>$	…	$<r_1,q_n>$	…
r_2	0.	$<r_2,q_1>$	$<r_2,q_2>$	$<r_2,q_3>$	$<r_2,q_4>$	…	$<r_2,q_n>$	…
r_3	0.	$<r_3,q_1>$	$<r_3,q_2>$	$<r_3,q_3>$	$<r_3,q_4>$	…	$<r_3,q_n>$	…
…								
…								
…								
r_n	0.	$<r_n,q_1>$	$<r_n,q_2>$	$<r_n,q_3>$	$<r_n,q_4>$	…	$<r_n,q_n>$	…
…								
…								
…								

We define operation $^+$ on digits: $0^+=1,\ldots,8^+=9$, $9^+ = 0$

Now we define a number **c** (for cantor) as follows:

c = 0.X, with X a countably infinite sequence of digits given by:

$$\forall n \in \mathbb{Z}^+\colon\ <\mathbf{c},q_n> = <r_n,q_n{}^+>$$

Hence:
if $<r_1,q_1> = 2$ then $<\mathbf{c},q_n> = 3$
if $<r_2,q_2> = 6$ then $<\mathbf{c},q_2> = 7$
if $<r_3,q_3> = 9$ then $<\mathbf{c},q_3> = 0$
…

		q_1	q_2	q_3	q_4	…	q_n	…
r_1	0.	**$<r_1,q_1>$**	$<r_1,q_2>$	$<r_1,q_3>$	$<r_1,q_4>$	…	$<r_1,q_n>$	…
r_2	0.	$<r_2,q_1>$	**$<r_2,q_2>$**	$<r_2,q_3>$	$<r_2,q_4>$	…	$<r_2,q_n>$	…
r_3	0.	$<r_3,q_1>$	$<r_3,q_2>$	**$<r_3,q_3>$**	$<r_3,q_4>$	…	$<r_3,q_n>$	…
…								
…								
…								
r_n	0.	$<r_n,q_1>$	$<r_n,q_2>$	$<r_n,q_3>$	$<r_n,q_4>$	…	**$<r_n,q_n>$**	…
…								
…								
…								

c:

	q_1	q_2	q_3	q_4	…	q_n	…
0.	**$\langle c,q_1^+ \rangle$**	$\langle r_1,q_2 \rangle$	$\langle r_1,q_3 \rangle$	$\langle r_1,q_4 \rangle$	…	$\langle r_1,q_n \rangle$	…
0.	$\langle r_2,q_1 \rangle$	**$\langle c,q_2^+ \rangle$**	$\langle r_2,q_3 \rangle$	$\langle r_2,q_4 \rangle$	…	$\langle r_2,q_n \rangle$	…
0.	$\langle r_3,q_1 \rangle$	$\langle r_3,q_2 \rangle$	**$\langle r_3,q_3^+ \rangle$**	$\langle r_3,q_4 \rangle$	…	$\langle r_3,q_n \rangle$	…
…							
…							
…							
0.	$\langle r_n,q_1 \rangle$	$\langle r_n,q_2 \rangle$	$\langle r_n,q_3 \rangle$	$\langle r_n,q_4 \rangle$	…	**$\langle r_n,q_n^+ \rangle$**	…
…							
…							
…							

Fact 1: c $\in (0,1)$.
c is 0. followed by a **countably** infinite sequence of digits.

Fact 2: c differs in place q_n from r_n.
This means that **c** differs from every number in the list r.
This means that **c** is not in the list r.
But, by hypothesis, r was a list of ***all*** real numbers in (0,1).
Hence this hypothesis is false: there is no such list.
Hence there are ***more*** real numbers than there are natural numbers.

In fact, you can prove:
Between any two rational numbers lie exactly as many numbers as there are rational numbers (i.e. countably many).
Between any two real numbers lie exactly as many numbers as there are real numbers.
(i.e. $2^{\aleph_0}$ many).

Linear orders and intervals.

We repeat:

Let **A** = $<A,\sqsubseteq>$ be a partial order, $a \in A$

a is a **minimal element in A** iff for no $b \in A$: $b \sqsubset a$
a is a **maximal element in A** iff for no $b \in A$: $a \sqsubset b$

a is **the minimum of A** iff for all $b \in A$: $a \sqsubseteq b$
a is **the maximum of A** iff for all $b \in A$: $b \sqsubseteq a$

If **A** has a minimum, we call the minimum 0_A (or just 0)
If **A** has a maximum, we call the maximum, 1_A (or just 1)

For the next definitions, we think our partial order **A** = $<A,\sqsubseteq>$ as running from left to right.

A is **right continuing** iff $\forall a \in A \exists b \in A$: $a \sqsubset b$
A is **left continuing** iff $\forall a \in A \exists b \in A$: $b \sqsubset a$
A is **continuing** iff **A** is left continuing and **A** is right continuing
A is **left-linear** iff for every $a, b_1, b_2 \in A$:
if $b_1 \sqsubset a$ and $b_2 \sqsubset a$ then $b_1 \sqsubset b_2$ or $b_2 \sqsubset b_1$ or $b_1=b_2$
A is **right-linear** iff for every $a, b_1, b_2 \in A$:
if $a \sqsubset b_1$ and $a \sqsubset b_2$ then $b_1 \sqsubset b_2$ or $b_2 \sqsubset b_1$ or $b_1=b_2$
A is **non-branching** iff **A** is left-linear and **A** is right-linear.

Let $<L,< >$ be a linear order.

A is **dense** iff for every $a_1`,a_2 \in A$: if $a_1 < a_2$ then $\exists b$: $a_1 < b < a_2$
A is **discrete** iff for every $a \in A$:
if $\exists b \in A$: $b < a$ then $\exists b \in A$: $b < a$ and $\neg\exists c$: $b < c < a$ and
if $\exists b \in A$: $a < b$ then $\exists b \in A$: $a < b$ and $\neg\exists c$: $a < c < b$ and

Every element that has a predecessor has an immediate predecessor and every element that has a successor has an immediate successor.
The natural numbers $< \mathbb{N}, < >$ form a discrete, right continuing linear order with minimum 0
Not every discrete, right continuing linear order with minimum 0 is isomorphic to the natural numbers.

Example
Let C (COPY) be an operation that maps a structure $<A, < >$ onto an isomorphic, non overlapping structure. Thus $C(<\mathbb{Z}, < >)$ is an isomorphic copy of the integers.
$<C(\mathbb{Z}), < >$ is a structure such that $\mathbb{Z} \cap C(\mathbb{Z}) = \emptyset$ (and, since $\mathbb{N} \subseteq \mathbb{Z}$, $N \cap C(\mathbb{Z}) = \emptyset$)
We define a new structure $\mathbb{N}$ + C($\mathbb{Z}$) = $<N \cup C(\mathbb{Z}), <_{N+C(Z)} >$
where $<_{\mathbb{N}+C(Z)}$ is defined as follows:

$$<_{\mathbb{N}+C(\mathbb{Z})} = <_{\mathbb{N}} \cup <_{C(\mathbb{Z})} \cup \{<a,b>: a \in \mathbb{N} \text{ and } b \in C(\mathbb{Z})\}$$

$\mathbb{N}$	0 1 2 3…
$\mathbb{N}$ + C($\mathbb{Z}$)	0 1 2 3......−3’ −2’ −1’ 0’ 1’ 2’ 3’ ….

$\mathbb{N}$ + C($\mathbb{Z}$) is a discrete, right continuing linear order with minimum 0, like **$\mathbb{N}$**.
But $\mathbb{N}$ and **$\mathbb{N}$ + C($\mathbb{Z}$)** are not isomorphic.

Proof: take a bijection f between **$\mathbb{N}$ + C($\mathbb{Z}$)** and **$\mathbb{N}$** and assume that $k \in C(\mathbb{Z})$. Then k has infinitely many predecessors, while f(k) has only finitely many predecessors. Then, obviously, it is not possible to preserve the order of all the predecessors of k into **ran**(f), because there are only finitely many values available. Hence f cannot preserve the order (i.e. for some $p <_{\mathbb{N}+C(Z)} k$: $\neg(f(p) <_N f(k))$.

The extra property that the natural numbers have is the second order property of *wellfoundedness*:

Let **A** = <A,⊑> be a partial order.

X is a **chain in** A iff X is a linear subset of A.
When not confusing, I will use **X** for <X, ⊑↾X>.
X is a **linear subset of A** iff $X \subseteq A$ and **X** is a linear order.

A is **wellfounded** iff every linear subset of **A** has a minimum.

Fact: A is isomorphic to **N** iff **A** is wellfounded, discrete, right continuing linear order with a minimum.

N + C($\mathbb{Z}$) is not wellfounded. $C(\mathbb{Z}) \subseteq N + C(\mathbb{Z})$, and C($\mathbb{Z}$) does not have a minimum.

A **tree** is a structure **T** = <T, ⊑, **top** > where <T,⊑ >is a wellfounded,discrete, left-linear order with minimum **top.**
Here the relation ⊑ is the **dominance relation**.

Note that in mathematics trees are not necessarily finite, and that if you want to impose a left-right order on the nodes of the tree you have to endow the structure with a second relation, a leftness relation.

A **left-right ordered tree** is a structure **T** = <T, ⊑, **top**, L > where:
2. <T, ⊑, **top** > is a tree.
3. L is a strict partial order on T such that:
for all $n_1,n_2 \in T$: (n_1 L n_2) or (n_2 L n_1) iff ($n_1 \not\sqsubseteq n_2$) and ($n_2 \not\sqsubseteq n_1$)

Fact: L satisfies **monotonicity**: If $a \sqsubseteq a_1$ and $b \sqsubseteq b_1$ and L(a,b), then L(a_1,b_1)

Let **A** = <A, ⊑ > be a partial order.

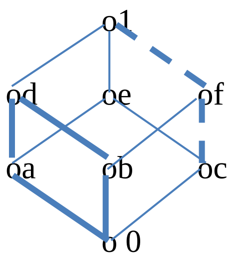
As we saw, a c**hain in A** is a linear subset of **A**.

X is a **maximal chain in A** iff X is a chain in **A** and for
every X ⊆ Y ⊆ A: if Y is a chain in **A** then X = Y.

A maximal chain in **A** is called a **path in A** or a **branch in A**.

Let X ⊆ A.
X is a **convex set** in **A** iff for every x_1,x_2 ∈ X: a ∈ A: if x_1 ⊑ a ⊑ x_2 then a ∈ X
X is an **interval in A** iff X is a **linear** convex set in **A**

A convex set is closed under untermediate elements.
Example: **A:**



{0,d} is not a convex subset of **A**, because 0,d, ∈ {0,d} and 0 ⊑ a ⊑ d but a ∉ {0,d}.
{0,a,b,d} is a convex subset of **A.**

Important note: {c,f,1} is **not** an interval in **A.**
The reason is that {c,f,1} is **not** a convex set in A. Namely, c,1 ∈ {c,f,1} and c ⊑ b ⊑ 1, but b ∉ {c,f,1}. So {c,f,1} is a linear subset of **A**, but it is not convex, hence not an interval in **A**.

If we want to express the sense in which {c,f,1} **is** an interval, we can do that by introducing a **branch in** A:

{c,f,1} and {0,c,f,1} are chains in **A**. {0,c,f,1} is a **branch in A** (adding any more element yields a set which is not linear anymore).
{c,f,1} is not an interval in **A**, but it **is** an interval in **{0,c,f,1}**.

We generalize the notions of join and meet:

Let **A** = <A,⊑> be a partial order. Let X ⊆ A and a ∈ A.

a is an **upper bound for** X iff for every x ∈ X: x ⊑ a
a is a **lower bound for** X iff for every x ∈ X: a ⊑ x

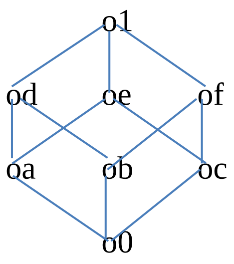
UB(X) is the set of all upper bounds for X.
LB(X) is the set of all lower bounds for X.
If **X** has a minimum,we call the minimum **min**(X) (we could also call it 0_X).
If **X** has a maximum, we call the maximum **max**(X).
Note that, by the notation convention, **min**(X) is the minimum in <X,⊑↾X>.

The **supremum of** X, $\sqcup$X, is:

$\sqcup X$ = **min**(UB(X)), the **lowest upper bound** of X

The **infimum of** X, $\sqcap$X, is:

$\sqcap X$ = **max**(LB(X)), the **greatest lower bound** of X



UB{a,b} = {d,1}	LB({a,b}) = {0}
min(UB({a,b})) = d	**max**({0}) = 0
$\sqcup$({a,b}) = d	$\sqcap$({a,b}) = 0

Note that $\sqcup\{a,b\} \notin \{a,b\}$ and $\sqcap\{a,b\} \notin \{a,b\}$

$\sqcup\{0,a,b,d\} = \sqcup\{a,b\} = d$

This time $\sqcup\{0,a,b,d\} \in \{0,a,b,d\}$ and $\sqcap\{0,a,b,d\} \in \{0,a,b,d\}$

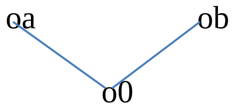
The notions of join and meet defined earlier are the two place variants:

$a \sqcup b = \sqcup(\{a,b\})$, $a \sqcap b = \sqcap(\{a,b,\})$

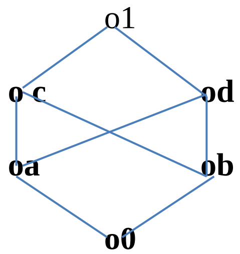
We call the general notions of join and meet also **complete join** and **complete meet**.

We repeat from above: Not for every partial order is it true that every subset has a supremum and an infimum.

B



C



In structure **B** {a,b} has an infimum (0), but not a supremum. UB({a,b}) = Ø, which does not have a minimum.

In **C** {a,b} also has an infimum (0) but not a supremum. This time UB({a,b}) = {c,1,d}, but again, {c,1,d} does not have a minimum. This time, {c,d} does not have an infimum, because LB({c,d}) = {a,b,0}, which does not have a maximum.

X is **upper bounded** iff X has a supremum

X is **lower bounded** iff X has an infimum

X is **bounded** iff X has both as supremum and an infimum.

Interval structures

If T is a linear order, an **interval in T** is a non-empty convex subset of T, a non-empty set which is closed under intermediate elements.

In a temporal context where we use intervals a lot and rely a lot on the notion of *bound*, it is useful to introduce *virtual bounds*.

A **linear order with virtual bounds** is a structure **T =** $\langle T \cup \{$ ¡$\infty, \infty\}, <,$ ¡$\infty, \infty\rangle$ where:

1. $\langle T, < \upharpoonright T\rangle$ is a linear order.
2. ¡$\infty, \infty \notin T$
3. for all $t \in T$: ¡$\infty < t < \infty$

We have defined above for $X \subseteq T$ the bounds of X (if they exist):
$\sqcap X$, the infimum (lower bound) of X and $\sqcup X$, the supremum (upper bound) of X.

A bound b for X is an **open bound** iff $b \notin X$
A bound b for X is a **closed bound** iff $b \in X$

If b is a closed lower bound for X, b is the **minimum** of X.
If be is a closed upper bound for X, b is the **maximum** of X.

We call a bounded set X an **open set** if the bounds are open, a **closed set** if the bounds are closed, and a **half open set** in the two other cases.

¡∞ and ∞ are virtual bounds: they are not in **T**, but are the lower resp. upper bounds of sets that do not have a lower resp. upper bound **in** T. Hence, while Z does not have a lower bound and an upper bound in Z, it does have a lower bound and an upper bound in Z with virtual bounds.
I will use the following terminology for $a,b \in T$

$[a,\rightarrow) = \{t \in T: a \leq t\}$	$[a,b] = \{t \in T: a \leq t \leq b\}$
$(a,\rightarrow) = \{t \in T: a < t\}$	$[a,b) = \{t \in T: a \leq t < b\}$
$(\leftarrow,a] = \{t \in T: t \leq a\}$	$(a,b] = \{t \in T: a < t \leq b\}$
$(\leftarrow,a) = \{t \in T: t < a\}$	$(a,b) = \{t \in T: a < t < b\}$

Note: I will use the notation $[a,\rightarrow)$ even if T has a maximum.
This interval notation is notation suitable only for **bounded** intervals.

Let $X \subseteq T$.
X is **left-extended** in T iff $\forall t \in T\ \exists x \in X: x < t$
X is **right-extended** in T iff $\forall t \in T\ \exists x \in X: t < x$

I will treat left or right extended subsets of T as bounded by the virtual bounds:

If X is left-extended in T then (by definition) $\sqcap(X) = -\infty$
If X is right-extended in T then (by definition) $\sqcup(X) = \infty$

In those cases I may also write $(-\infty,a]$ for $(\leftarrow,a]$.

Note : An important subtlety: I define intervals as subsets of T, excluding the virtual bounds. This means that an interval bounded by virtual bounds is an **open** interval. Note too that intervals, on this definition, are not necessarily bounded.

Let **T** = $<T, <_{\mathbf{T}}>$ be a linear order (with virtual bounds).

The **interval structure of T**, $\mathbf{I_T} = <I_T, \subseteq, <>$ where

1. I_T is the set of all intervals in T
2. $\subseteq$, the inclusion relation, is the subset relation on I_T.
3. $<$, the relation of precedence, is defined on $I_T - \{\emptyset\}$ by:

   for all $i,j \in I_T$: $i < j$ iff $\forall t_1 \in i\ \forall t_2 \in j$: $t_1 <_T t_2$

The reason we define $<$ on $I_T - \{\emptyset\}$ is that we want $<$ to be a linear order on intervals in a sense defined below. If we include $\emptyset$, $<$ is not irreflective (since it would be true that $\emptyset < \emptyset$).

The interval structures defined here are also called point-generated interval structures.

We can define overlap in terms of inclusion:

i O j, i **overlaps** j iff $\exists k \in I_T$: $k \subseteq i$ and $k \subseteq j$

Note that we require that i and j overlap in an **interval**, a **non-empty** convex subset. Don't think here in terms of subsets: the empty set does not count as a 'real' subinterval.

With precedence and overlap we define the intuitive notion of linearity for intervals:

$<$ is **i-linear** iff for all $i,j \in I_T - \{\emptyset\}$: $i < j$ or $j < i$ or i O j

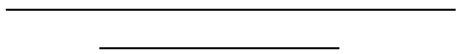
Fact: If T is linear, I_T is i-linear.

We can define a variety of other useful notions:

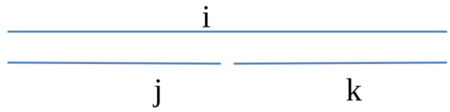
i and j **co-start** iff $\forall k \in I_T$: $k < i$ iff $k < j$

i and j **co-end** iff $\forall k \in I_T$: $i < k$ iff $j < k$

i **is an initial subinterval of** j iff $i \subseteq j$ and i and j co-start
i **is a final subinterval of** j iff $i \subseteq j$ and i and j co-end
i **is a medial subinterval of** j iff $i \subseteq j$ and neither i and j co-start
and neither i and j co-end



i **splits into** j and k, $i = j + k$ iff $i = j \cup k$ and $j < k$



This means that i **partitions** into two intervals j and k, with j before k.

i is a **point-interval** in I_T iff $i = \{t\}$ for some $t \in T$.

$POINT_I$ is the set of all point intervals in $\mathbf{I_T}$.

A **cut through** i is a pair of intervals $<j,k>$ such that $j + k = i$

Let $<j,k>$ be a cut through T.

$<j,k>$ detemines a **jump** iff $\sqcup j \in j$ and $\sqcap k \in k$
$<j,k>$ determines a **transition** iff $\sqcap k \in j$ or $\sqcup j \in k$
$<j,k>$ determines a **gap** iff $\sqcup j = \bot$ and $\sqcap k = \bot$
(j has no upper bound and k has no lower bound)

Let $<T,<>$ be a linear order.

Lemma 1: T is **dense** iff no cut through T determines a jump
Proof.
-Suppose $<i,j>$ is a cut through T that determines a jump.
Then $\sqcup i \in i$ and $\sqcap j \in j$.
By definition of cut, $\{i,j\}$ is a partition of T and $\sqcup i < \sqcap j$.
But then $\neg\exists t \in T$: $\sqcup i < t < \sqcap j$, and hence T is not dense.
-Suppose that T is not dense. Then for some $t_1,t_2 \in T$ $t_1 < t_2$ and $\neg\exists t \in T$: $t_1 < t < t_2$,
Then $<(\leftarrow,t_1],[t_2,\rightarrow)>$ is a cut through T that determines a jump.

Note: remember that the notation $(\leftarrow,t_1)$ does not mean that the lower bound of $(\leftarrow,t_1)$ is a virtual bound, or that $(\leftarrow,t_1)$ is an open interval. The subinterval $(\leftarrow,10)$ of $\mathbf{N}$ is $[0,10)$.
On the other hand $(-\infty,a)$ **is** an open interval.

Lemma 2: T is **discrete** iff no cut through T determines a transition
Proof.
Similar.

We define:

T is **continuous** iff no cut through T determines a gap.

-Continuity is the same as completeness: all sets have bounds.
A gap is a situation of two sets approximating each other without a bound, like the cut:
$<\{q \in \mathbf{Q}: q < \pi\}, \{q \in \mathbf{Q}: \pi < q\}>$, where $\pi \notin \mathbf{Q}$
We can also express a principle like density directly at the interval structure. In that case, we formulate it as a property of intervals bigger than a point.

Let us call a interval bigger than a point a **proper interval**.

$\mathbf{I_T}$ is **dense** iff for every $i \in I_T$ ¡ $POINT_I$: $\exists j,k \in I_T$ ¡ $POINT_I$: $i = j + k$

Every proper interval in I_T can be split into proper intervals.

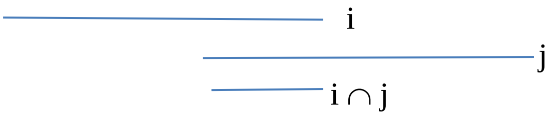
Note that if T is discrete, T is continuous. But, as we will see, there are continuous orders that are not discrete, in fact, dense continuous orders.

Some more operations on intervals.

Let I_T be an interval structure.

Lemma: $\mathbf{I_T}$ is closed under non-empty intersection:

If $i, j \in I_T$ and $i \cap j \neq \emptyset$, then $i \cap j \in I_T$



Proof
It is easy to prove that the non-empty intersection of two intervals is itself an interval.

The interval structure **I** is not closed under union, and not under complementation.
-the union of Yesterday and Tomorrow is not itself an interval.
-The complement of Today, is all time, except today, which is not an interval.
Both of these are not convex sets.

Let $X \subseteq T$ and $Z \subseteq I_T$

The **convex closure of** X, X^{cc},

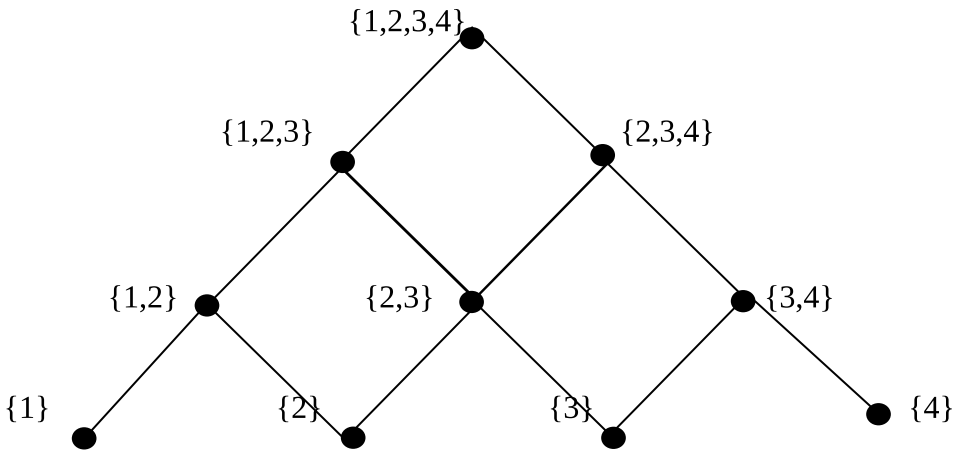
$\mathbf{X^{cc}} = \{t \in T: \exists t_1 \in X\ \exists t_2 \in X: t_1 \leq t \leq t_2\}$

The **convex union** of Z, $\cup^c(Z)$,

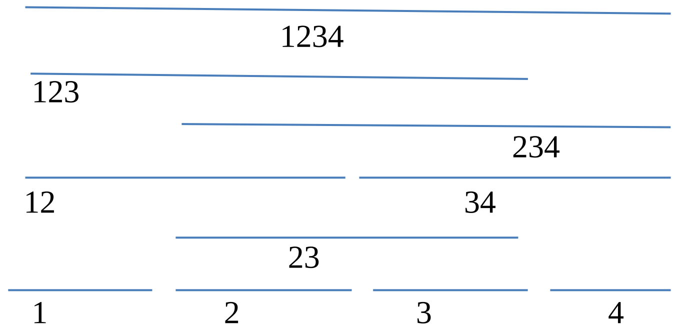
$\cup^c(Z) = (\cup Z)^{cc}$

Lemma: $\mathbf{I_T}$ is closed under convex union

The diagrams of interval structures are *half-chess boards*:



Or, in another visualization:



Convex union is the supremum operation under ⊆:
In this structure, the join operation is convex union,
so: {1} $\cup^c$ {4} = {1,2,3,4}

Linear orders, discrete, dense, and continuous

In a discrete linear order no cut determines a transition; in a continuous linear order no cut determines a gap. Hence in a discrete continuous linear order every cut determines a jump.

Theorem 1. If <A,< > is a discrete continuous linear order then <A,< > is isomorphic to a substructure of < ℤ, < >

Proof.
Let a,b ∈ A and a < b.
Since every cut in A determines a jump, there is no transition or gap between a and b.
This means that there are finitely many elements between a and b.
This means that A is at most countable.
You make an isomorphism with a substructure of ℤ as follows.
Pick any element a ∈ A. Choose f(a) = 0.
If for some b ∈ A a < b then a has a direct successor in A a+1. Set f(a+1) = 1
If for some b ∈ A b < a then a has a direct predecessor in A a–1. Set f(a–1) = –1
Since A is a linear order for any element b ∈ A a < b or b < a. hence b ∈ A comes into the function after a finite number of steps. The function f obviously preserves the order.

Hence up to isomorphism the discrete continuous linear orders are the finite linear orders, $\mathbb{Z}^+$, $\mathbb{Z}^-$ and ℤ.

Let $[a,b]_{\mathbb{Q}}$ be the closed subinterval [a,b] of ℚ.

Theorem 2: Cantor's Theorem

Every countable dense linear order is isomorphic to one of
<$\mathbf{[0,1]}_{\mathbb{Q}}$,< >, <$\mathbf{[0,1)}_{\mathbb{Q}}$,< >, <$\mathbf{(0,1]}_{\mathbb{Q}}$,< >, <$\mathbf{(0,1)}_{\mathbb{Q}}$,< >

We only need to prove one of these:

Every continuing countable dense linear is isomorphic to <$\mathbf{(0,1)}_{\mathbb{Q}}$,< >
(and hence to **ℚ**).

Proof.
Let <A,< > and <B,< > be continuing countable dense linear orders.
Let *A* = $a_0,a_1,...,a_n,...$ be an enumeration of A, and *B* = $b_0,b_1,...,b_n,...$ be an emumeration of B.
Since A and B are countable, such enumerations exist.

We define a sequence $f_0,f_1,....f_n,...$ as follows:
1. $f_0 = \{<a_0,b_0>\}$
Note that f_0 is, trivially, a finite one-one function that preserves the order.

2. If n>0 and n is **odd**, then $f_n = f_{n-1} \cup \{<a,b>\}$ where:

2a: a is the first element in enumeration *A* such that a ∉ **dom**(f_{n-1}).
Since **dom**(f_{n-1}) is finite, and A countable there always is a first element in *A* not in **dom**(f_{n-1}).

$2b_1$: If for every x ∈ **dom**(f_{n-1}): a < x, then b is the first element in enumeration *B* such that b∉ **ran**(f_{n-1}) and for every y ∈ **ran**(f_{n-1}): b < y.

Since **ran**(f_{n-1}) is finite, and B countable, and since B is continuing,
there always is a first element in *B* not in **ran**(f_{n-1}) and before every element in **ran**(f_{n-1}).

2b2: If for every $x \in$ **dom**(f_{n-1}): $x < a$, then b is the first element in enumeration *B* such that $b \notin$ **ran**(f_{n-1}) and for every $y \in$ **ran**(f_{n-1}): $y < b$.
Since **ran**(f_{n-1}) is finite, and B countable, and since B is continuing,
there always is a first element in *B* not in **ran**(f_{n-1}) and after every element in **ran**(f_{n-1}).

Since A is linear, if $a \notin$ **dom**(f_{n-1}) and a is not before every x in **dom**(f_{n-1}) and not after every x in **dom**(f_{n-1}), then for some $x_1,x_2 \in$ **dom**(f_{n-1}): $x_1 < a < x_2$.
Since **dom**(f_{n-1}) is finite this means that for some $x_1,x_2 \in$ **dom**(f_{n-1}): $x_1 < a < x_2$
and for no $x_3 \in$ **dom**(f_{n-1}): $x_1 < x_3 < a$ and for no $x_3 \in$ **dom**(f_{n-1}): $a < x_3 < x_2$.

2b3: If for some $x_1,x_2 \in$ **dom**(f_{n-1}): $x_1 < a < x_2$
and for no $x_3 \in$ **dom**(f_{n-1}): $x_1 < x_3 < a$ and for no $x_3 \in$ **dom**(f_{n-1}): $a < x_3 < x_2$,
then b is the first element in *B* such that $b \notin$ **ran**(f_{n-1}) and $f_{n-1}(x_1) < b < f_{n-1}(x_2)$.
Since **ran**(f_{n-1}) is finite, and B countable, and since B is dense, there always is a first element in *B* not in **ran**(f_{n-1}) and between $f_{n-1}(x_1)$ and $f_{n-1}(x_2)$.

Note that, by the construction, if f_{n-1} is a finite one-one function that preserves the order, then so is f_n. We add to f_{n-1} one pair $< a,b >$, where, by the construction, $< a,b >$ is well defined, $a \notin$ **dom**(f_{n-1}), $b \notin$ **ran**(f_{n-1}).
This means that, by the construction, if f_{n-1} is a function, so is f_n; if f_{n-1} is one-one, so is f_n, and if f_{n-1} preserves the order, so does f_n.

3. If $n>0$ and n is **even**, then $f_n = f_{n-1} \cup \{< a,b >\}$ where:

3a: b is the first element in enumeration *B* such that $b \notin$ **ran**(f_{n-1}).
Since **ran**(f_{n-1}) is finite, and B countable there always is a first element in *B* not in **ran**(f_{n-1}).

3b1: If for every $y \in$ **ran**(f_{n-1}): $b < y$, then a is the first element in enumeration *A* such that $a \notin$ **dom**(f_{n-1}) and for every $x \in$ **dom**(f_{n-1}): $a < x$.
Since **dom**(f_{n-1}) is finite, and A countable, and since A is continuing,
there always is a first element in *A* not in **dom**(f_{n-1}) and before every element in **dom**(f_{n-1}).

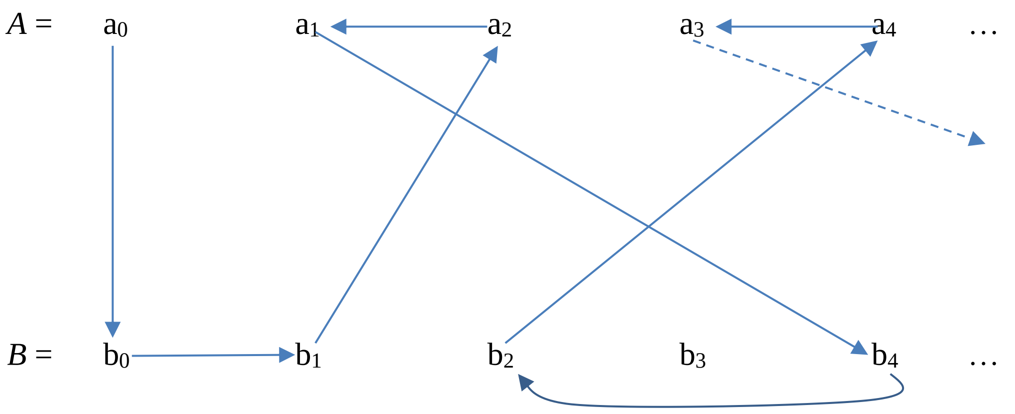
3b2: If for every $y \in$ **ran**(f_{n-1}): $y < b$, then a is the first element in enumeration *A* such that $a \notin$ **dom**(f_{n-1}) and for every $x \in$ **dom**(f_{n-1}): $x < a$.
Since **dom**(f_{n-1}) is finite, and A countable, and since A is continuing,
there always is a first element in *A* not in **dom**(f_{n-1}) and after every element in **dom**(f_{n-1}).
Since B is linear, if $b \notin$ **ran**(f_{n-1}) and b is not before every y in **ran**(f_{n-1}) and not after every y in **ran**(f_{n-1}), then for some $y_1,y_2 \in$ **ran**(f_{n-1}): $y_1 < b < y_2$. Since **ran**(f_{n-1}) is finite this means that for some $y_1,y_2 \in$ **ran**(f_{n-1}): $y_1 < b < y_2$
and for no $y_3 \in$ **ran**(f_{n-1}): $y_1 < y_3 < b$ and for no $y_3 \in$ **ran**(f_{n-1}): $b < y_3 < y_2$.

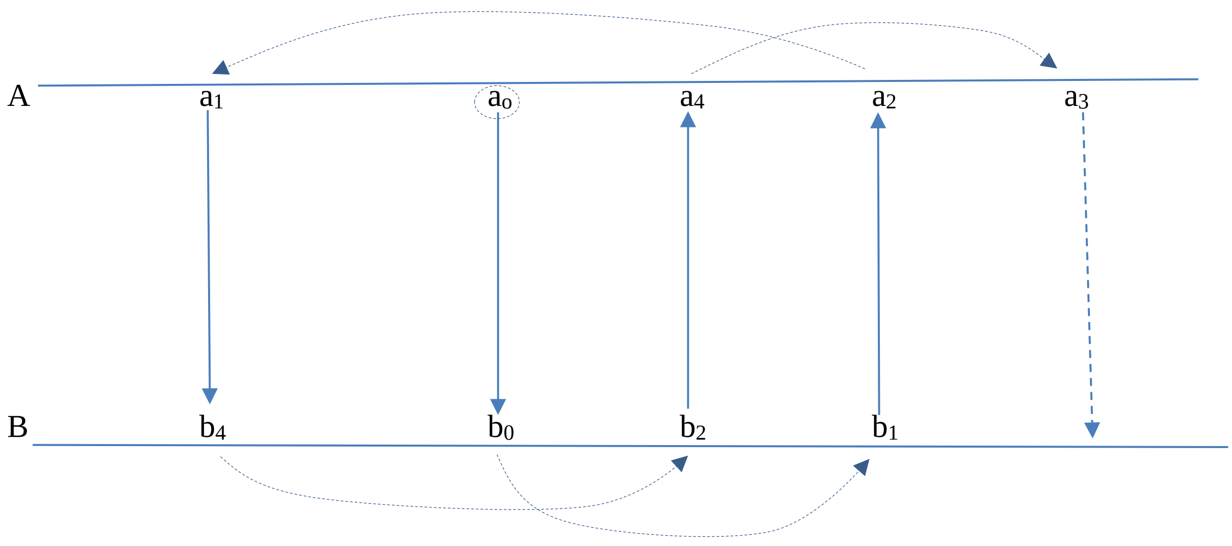
3b3: If for some $y_1,y_2 \in$ **ran**(f_{n-1}): $y_1 < b < y_2$
and for no $y_3 \in$ **ran**(f_{n-1}): $y_1 < y_3 < b$ and for no $y_3 \in$ **ran**(f_{n-1}): $b < y_3 < y_2$, then a is the first element in *A* such that $a \notin$ **dom**(f_{n-1}) and $f_{n-1}^{-1}(y_1) < a < f_{n-1}^{-1}(y_2)$.

Since **dom**(f_{n-1}) is finite, and A countable, and since A is dense, there always is a first element in *A* not in **dom**(f_{n-1}) and between $f_{n-1}^{-1}(y_1)$ and $f_{n-1}^{-1}(y_2)$.

Note that, by the construction, if f_{n-1} is a finite one-one function that preserves the order, then so is f_n.
We add to f_{n-1} one pair <a,b>, where, by the construction, <a,b> is well defined, $a \notin$ **dom**(f_{n-1}), $b \notin$ **ran**(f_{n-1}). This means that, by the construction, if f_{n-1} is a function, so is f_n; if f_{n-1} is one-one, so is f_n, and if f_{n-1} preserves the order, so does f_n.

The construction in a picture:





Fact 1: For every n: f_n is a finite one-one function which preserves the order.
Proof: The induction steps are given in the construction.

Fact 2: For every $a \in$ A there is an n such that $a \in$ **dom**(f_n).
For every $b \in$ B there is an n such that $b \in$ **ran**(f_n).

Proof: This follows from the zig-zag construction.
If $a \in$ A and for some k, $a \notin$ **dom**(f_k), then for some m, a is the m-th element of *A* not in **dom**(f_k). By the construction, this means that $a \in$ **dom**(f_{k+2m}), either because it is chosen as

the argument for some b before that, or, if not, because at that stage it is the first element in the enumeration *A* which isn't in the domain of the previous function.
The very same argument applies to any $b \in B$.

Now we define:

$$f = \cup\{f_n: n \geq 0\}$$

Fact 3: f is an isomorphism between <A,< > and <B,< >.

Proof.
-f is, of course, by definition a relation between A and B.
-Since each f_{n+1} is a function extending f_n, f is a function.
-By definition, **dom**(f) = $\cup\{$**dom**$(f_n): n \geq 0\}$. By fact 2, this is A.
Thus f is a function from A into B.
-By definition, **ran**(f) = $\cup\{$**ran**$(f_n): n \geq 0\}$. By fact 2, this is B.
Thus f is a function from A onto B.
-Since each f_n is a one-one function, f is a one-one function. If $a_1,a_2 \in$ **dom**(f) and $f(a_1)=f(a_2)$, then for some n: $a_1,a_2 \in$ **dom**(f_n), and hence $f_n(a_1)=f_n(a_2)$,. But then, since f_n is one-one, $a_1=a_2$.
Thus f is a bijection between A and B.
-Since each f_n preserves the order, f preserves the order. If $a_1 < a_2$, then, since for some n, $a_1,a_2 \in$ **dom**(f_n), by construction $f_n(a_1) < f_n(a_2)$. But then $f(a_1) < f(a_2)$.
□

We have so far dealt with linear orders with only jumps and with countable linear orders without jumps.
What about linear orders with only gaps, and linear orders with only transitions?

Concerning the first, it is easy to see that they do not exist.
Namely, let A be any linear order and let $a \in A$, but not an endpoint. Then $<(\leftarrow,b],(b,\rightarrow)>$ determines a jump or a transition. This means that it can't be the case that every cut in A determines a gap.

This means that for linear orders without jumps (i.e. dense linear orders) we can find only two possible kinds: with gaps and transitions, or with only transitions.
The countable cases are cases with gaps and transitions. Gaps in $\mathbb{Q}$ can be shown by looking at irrational numbers. let $r \in \mathbb{R} - \mathbb{Q}$:
$\mathbb{Q} = (\leftarrow,r) \cup (r,\rightarrow)$, hence $<(\leftarrow,r) ,(r,\rightarrow)>$ is a cut through $\mathbb{Q}$ that determines gap.

Another nice way of showing that some cut in $\mathbb{Q}$ determines a gap is by considering $\mathbb{Q} + C(\mathbb{Q})$, $\mathbb{Q}$ with a copy of itself after it. Obviously, $< \mathbb{Q}, C(\mathbb{Q})>$ determines a gap in $\mathbb{Q} + C(\mathbb{Q})$.
But $\mathbb{Q} + C(\mathbb{Q})$ is itself a continuing countable dense linear order, and hence by Cantor's theorem isomorphic to $\mathbb{Q}$. So some cut in $\mathbb{Q}$ determines a gap.

Even more dramatic: replace every element of $\mathbb{Q}$ by a full copy of $\mathbb{Q}$, preserving the order.
This structure is also isomorphic to $\mathbb{Q}$.

It follows that linear orders in which every cut determines a transition can only be non-countable. Intuitively we get the set of real numbers $\mathbb{R}$ by for every cut in $\mathbb{Q}$ that determines a gap, filling up the gap with an irrational number. This turns the gap into a transition (because you need to extend either T_1 or T_2 of the original cut which determined a gap with the element added to get a partition). As it turns out, there is a way of doing this, and in fact only one way of doing this.

Let <**B**,< > be a continuing dense linear order, and let $A \subseteq B$.

A lies **dense in** B iff for every $b_1,b_2 \in B$: if $b_1 < b_2$ then there is an $a \in A$: $b_1 < a < b_2$.

Theorem 3: Let <B,< > be a continuing linear order in which every cut determines a transition, and A a **countable** subset of B which lies **dense** in B.
Then <B,<> is isomorphic to < $\mathbb{R}$,< >

Rationale.
If <A,< > is a continuing linear order in which every cut determines a transition and a countable subset B lies dense in A, then every transition can be reconstructed as the bounds of a cut of intervals in B, and A is just the result of **adding** these bounds where they are lacking in B (when there are gaps).
Such a structure is called a **completion** of the structure < $\mathbb{Q}$,< >.
You can prove that each incomplete structure has (up to isomorphism) one and only completion, and that the completions of isomorphic incomplete structures are themselves isomorphic. From this the result follows.

Note that not every continuing linear order in which every cut determines a transition is isomorphic $\mathbb{R}$. Only those that have a countable subset which is dense in them.