LUND UNIVERSITY

EDAP15: Program Analysis

DATAFLOW ANALYSIS 2: VIDEOS

Christoph Reichenbach

Data Flow Analysis on CFGs

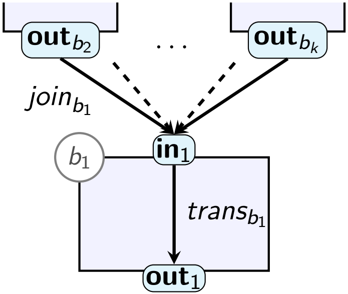
- $join_b$: *Join Function*
- $trans_b$: *Transfer Function*
- $\mathbf{in}_b$:

$$\mathbf{in}_{b_1} = join_{b_1}(\mathbf{out}_{b_2}, \ldots, \mathbf{out}_{b_k})$$

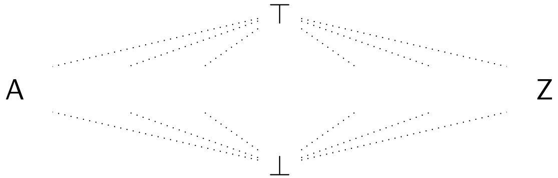
- $\mathbf{out}_b$:

$$\mathbf{out}_{b_1} = trans_{b_1}(\mathbf{in}_{b_1})$$

- *Forward Analysis*
- *Bakward Analysis*

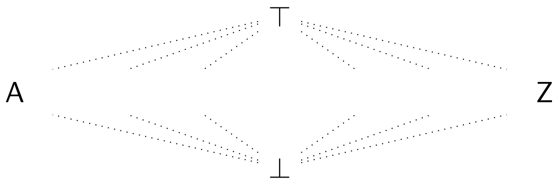


Join and Transfer Functions



- L: Abstract Domain
  - Ordered by $(\sqsubseteq) \subseteq L \times L$

Join and Transfer Functions

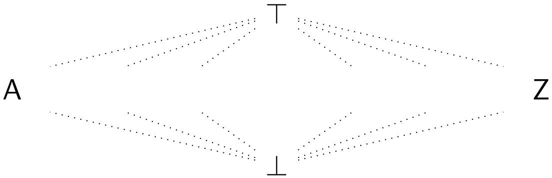


- L: Abstract Domain
  - Ordered by $(\sqsubseteq) \subseteq L \times L$

    $\top \in L$ for all x : $x \sqsubseteq \top$ Top element

    $\bot \in L$ for all x : $\bot \sqsubseteq x$ Bottom element (optional)

Join and Transfer Functions



- L: Abstract Domain
  - Ordered by $(\sqsubseteq) \subseteq L \times L$

    $\top \in L$ for all x : $x \sqsubseteq \top$ Top element

    $\bot \in L$ for all x : $\bot \sqsubseteq x$ Bottom element (optional)
- $trans_b : L \to L$
  - monotonic

$$\begin{array}{ccc} x & \sqsubseteq & y \\ & \Downarrow & \\ trans_b(x) & \sqsubseteq & trans_b(y) \end{array}$$

- $join_b : L \times \ldots \times L \to L$
  - pointwise monotonic

$$\begin{array}{ccc} x & \sqsubseteq & y \\ & \Downarrow & \\ join_b(z_1, \ldots, z_k, x, \ldots, z_n) & \sqsubseteq & join_b(z_1, \ldots, z_k, y, \ldots, z_n) \end{array}$$

Lattices (‘gitter’ in Swedish)

Image by Emma Mae (Flickr) via Wikimedia commons

Partially Ordered Set

Lattices L are based on a *partially ordered set* $\langle \mathcal{L}, \sqsubseteq \rangle$:

- Set: $\mathcal{L}$ describes possible information
- $(\sqsubseteq) \subseteq \mathcal{L} \times \mathcal{L}$:
- Intuition for $a \sqsubseteq b$ (for program analysis):
  - b has at least as much information as a

Partially Ordered Set

Lattices L are based on a *partially ordered set* $\langle \mathcal{L}, \sqsubseteq \rangle$:

- Set: $\mathcal{L}$ describes possible information
- $(\sqsubseteq) \subseteq \mathcal{L} \times \mathcal{L}$:
- Intuition for $a \sqsubseteq b$ (for program analysis):
  - b has at least as much information as a
- $(\sqsubseteq)$ is a *partial order*:

$$
\begin{array}{ll}
a \sqsubseteq a & \textit{Reflexivity} \\
a \sqsubseteq b \text{ and } b \sqsubseteq a \implies a = b & \textit{Antisymmetry} \\
a \sqsubseteq b \text{ and } b \sqsubseteq c \implies a \sqsubseteq c & \textit{Transitivity}
\end{array}
$$

Partially Ordered Set

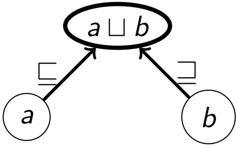
Lattices L are based on a *partially ordered set* $\langle \mathcal{L}, \sqsubseteq \rangle$:

- Set: $\mathcal{L}$ describes possible information
- $(\sqsubseteq) \subseteq \mathcal{L} \times \mathcal{L}$:
- Intuition for $a \sqsubseteq b$ (for program analysis):
  - b has at least as much information as a
- $(\sqsubseteq)$ is a *partial order*:

$$\begin{array}{lll} a \sqsubseteq a & & \textit{Reflexivity} \\ a \sqsubseteq b \text{ and } b \sqsubseteq a & \Longrightarrow \; a = b & \textit{Antisymmetry} \\ a \sqsubseteq b \text{ and } b \sqsubseteq c & \Longrightarrow \; a \sqsubseteq c & \textit{Transitivity} \end{array}$$

- Example:
  - $\mathcal{L} = \{\textit{unknown}, \textit{true}, \textit{false}, \textit{true-or-false}\}$
  - $\textit{unknown} \sqsubseteq \textit{true} \sqsubseteq \textit{true-or-false}$
  - $\textit{unknown} \sqsubseteq \textit{false} \sqsubseteq \textit{true-or-false}$

Least Upper Bound

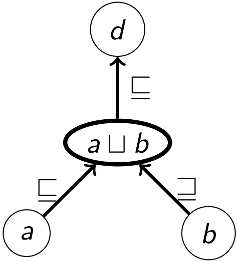


Combining potentially contradictory information:

- *Join operator*: $(\sqcup) : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$
- Pointwise monotonic:

$$a \sqsubseteq a \sqcup b \text{ and } b \sqsubseteq a \sqcup b$$

Least Upper Bound



Combining potentially contradictory information:

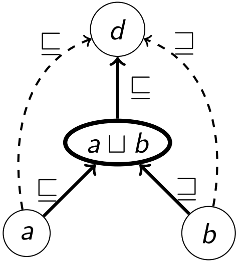
- *Join operator*: $(\sqcup) : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$
- Pointwise monotonic:

$$a \sqsubseteq a \sqcup b \text{ and } b \sqsubseteq a \sqcup b$$

- *Least* element with this property:

$$a \sqsubseteq d \text{ and } b \sqsubseteq d \implies a \sqcup b \sqsubseteq d$$

Least Upper Bound



Combining potentially contradictory information:

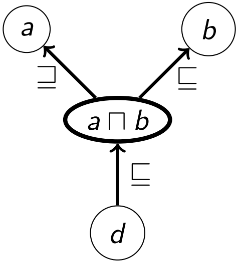
- *Join operator*: $(\sqcup) : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$
- Pointwise monotonic:

$$a \sqsubseteq a \sqcup b \text{ and } b \sqsubseteq a \sqcup b$$

- *Least* element with this property:

$$a \sqsubseteq d \text{ and } b \sqsubseteq d \implies a \sqcup b \sqsubseteq d$$

Greatest Lower bound



Converse operation:

- *Meet operator*: $(\sqcap) : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$
- Pointwise monotonic:

$$a \sqcap b \sqsubseteq a \text{ and } a \sqcap b \sqsubseteq b$$

- *Greatest* element with this property:

$$d \sqsubseteq a \text{ and } d \sqsubseteq b \implies d \sqsubseteq a \sqcap b$$

Lattices

$$L = \langle \mathcal{L}, \sqsubseteq, \sqcap, \sqcup \rangle$$

- $\mathcal{L}$: Underlying set
- $(\sqsubseteq) \subseteq \mathcal{L} \times \mathcal{L}$: Partial Order
- $(\sqcup) : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$: Join (computes l.u.b.)
- $(\sqcap) : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$: Meet (computes g.l.b.)

Lattices

$$L = \langle \mathcal{L}, \sqsubseteq, \sqcap, \sqcup \rangle$$

- $\mathcal{L}$: Underlying set
- $(\sqsubseteq) \subseteq \mathcal{L} \times \mathcal{L}$: Partial Order
- $(\sqcup) : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$: Join (computes l.u.b.)
- $(\sqcap) : \mathcal{L} \times \mathcal{L} \to \mathcal{L}$: Meet (computes g.l.b.)

- can show:

$$\begin{aligned} \text{Commutativity:} \quad a \sqcup b &= b \sqcup a \\ \text{Associativity:} \quad a \sqcup (b \sqcup c) &= (a \sqcup b) \sqcup c \end{aligned}$$

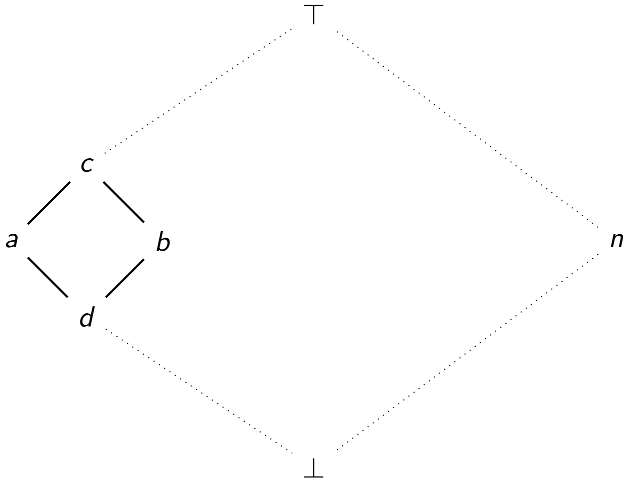
(Analogous for $\sqcap$)

Complete Lattices

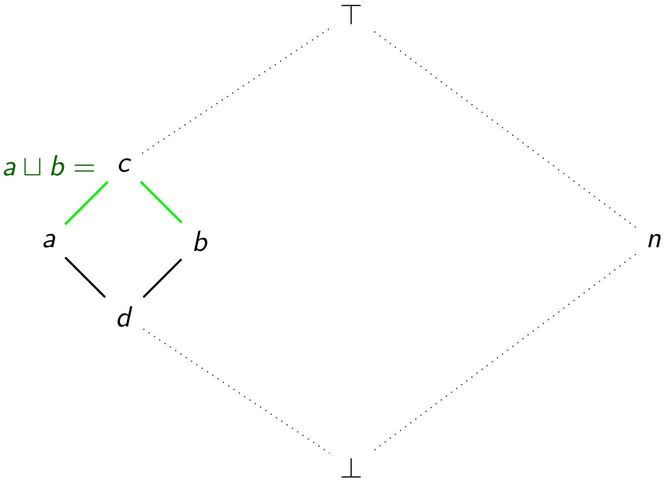
A lattice $L = \langle \mathcal{L}, \sqsubseteq, \sqcap, \sqcup \rangle$ is *complete* iff:

- For any $\mathcal{L}' \subseteq \mathcal{L}$ there exist:
  - $\top = \bigsqcup \mathcal{L}'$
  - $\bot = \bigsqcap \mathcal{L}'$

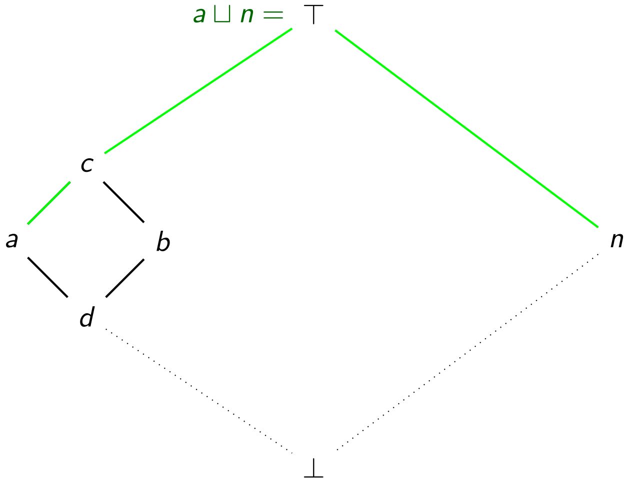
Complete Lattices: Visually



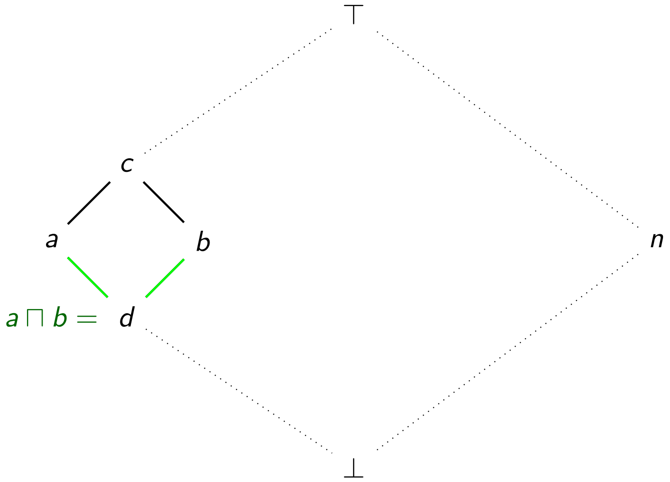
Complete Lattices: Visually



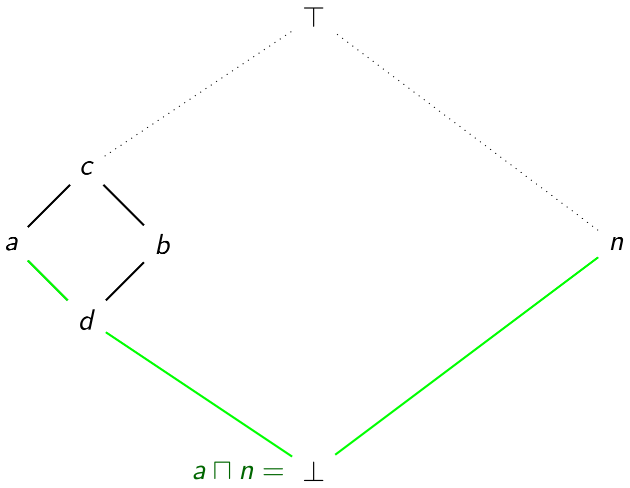
Complete Lattices: Visually



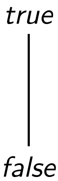
Complete Lattices: Visually



Complete Lattices: Visually

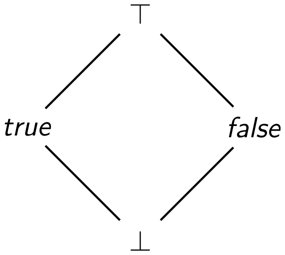


Example: Binary Lattice



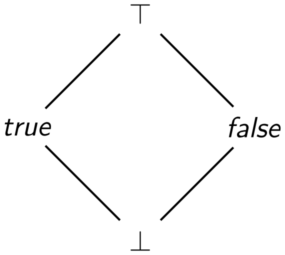
- $\top =$ *true*
- $\bot =$ *false*
- $\sqcup =$ *logical "or"*
- $\sqcap =$ *logical "and"*

Example: Booleans



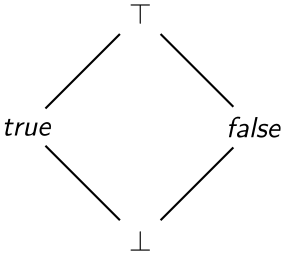
- If $\mathbb{B} = \{true, false\}$:
  - Lattice sometimes called $\mathbb{B}_\bot^\top$

Example: Booleans



- If $\mathbb{B} = \{true, false\}$:
  - Lattice sometimes called $\mathbb{B}_{\bot}^{\top}$
- Interpretation for data flow e.g.:
  - $\top$ = true-or-false
  - $\bot$ = unknown
  - $a \sqcup b$: either a or b
  - $a \sqcap b$: both a and b

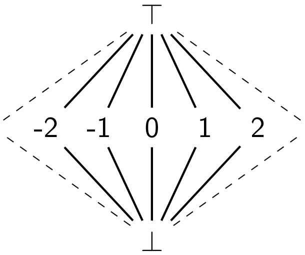
Example: Booleans



- If $\mathbb{B} = \{true, false\}$:
  - Lattice sometimes called $\mathbb{B}_{\bot}^{\top}$
- Interpretation for data flow e.g.:
  - $\top$ = true-or-false
  - $\bot$ = unknown
  - $a \sqcup b$: either a or b
  - $a \sqcap b$: both a and b

Other interpretations possible

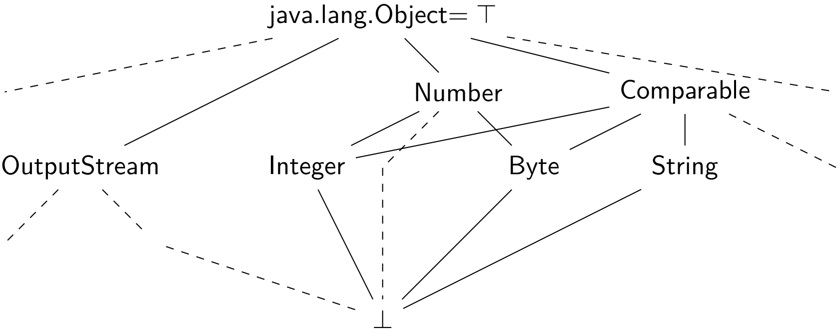
Example: Flat Lattice on Integers



- Sometimes written $\mathbb{Z}_\bot^\top$
- $\top = \mathbb{Z}$
- $\bot = \emptyset$
- $a \sqcup b = \begin{cases} a & \text{iff} \quad a = b \\ \top & \quad\text{otherwise} \end{cases}$
- $a \sqcap b = \begin{cases} a & \text{iff} \quad a = b \\ \bot & \quad\text{otherwise} \end{cases}$

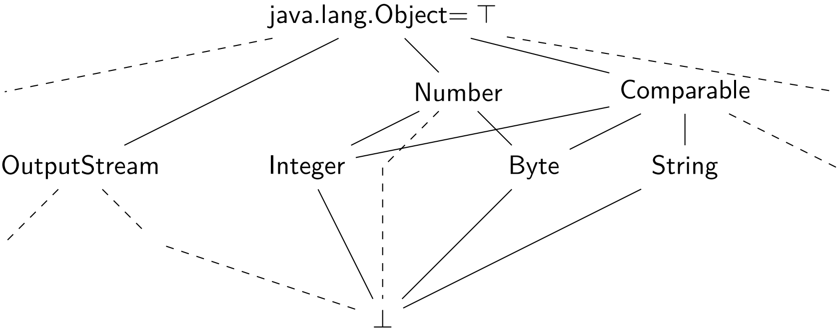
Analogous for other $X_\bot^\top$ from set X

Example: Type Hierarchy Lattices



- $\sqcup$ constructs most precise supertype

Example: Type Hierarchy Lattices



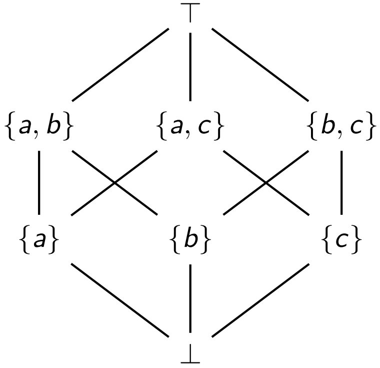
- $\sqcup$ constructs most precise supertype
- $\sqcap$ constructs *intersection types*:

$$\text{java.lang.Comparable} \sqcap \text{java.io.Serializable}$$

- *Java notation*:

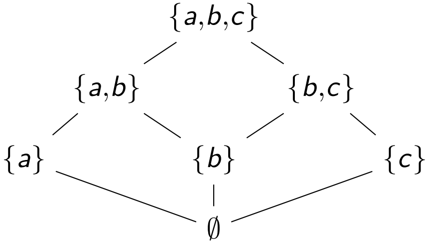
```
java.lang.Comparable & java.io.Serializable
```

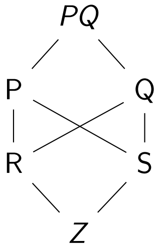
Example: Powersets



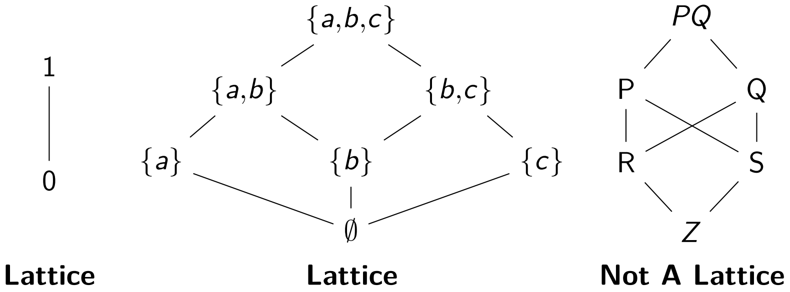
- Take any set $S = \{a, b, c\}$
- $\mathcal{L} = \mathcal{P}(S)$
- $\top = S$
- $\bot = \emptyset$
- $(\sqcup) = (\cup)$
- $(\sqcap) = (\cap)$

Example: Lattices and Non-Lattices





Example: Lattices and Non-Lattices



Right-hand side is missing e.g. a unique $R \sqcup S$

Example: Natural numbers with 0, ω

- $\top = \omega$
- $\bot = 0$
- $a \sqcup b =$ maximum of a and b
- $a \sqcap b =$ minimum of a and b

Product Lattices

- Assume (complete) lattices:
  - $L_1 = \langle \mathcal{L}_1, \sqsubseteq_1, \sqcap_1, \sqcup_1, \top_1, \bot_1 \rangle$
  - $L_2 = \langle \mathcal{L}_2, \sqsubseteq_2, \sqcap_2, \sqcup_2, \top_2, \bot_2 \rangle$

Product Lattices

- Assume (complete) lattices:
  - $L_1 = \langle \mathcal{L}_1, \sqsubseteq_1, \sqcap_1, \sqcup_1, \top_1, \bot_1 \rangle$
  - $L_2 = \langle \mathcal{L}_2, \sqsubseteq_2, \sqcap_2, \sqcup_2, \top_2, \bot_2 \rangle$
- Let $L_1 \times L_2 = \langle \mathcal{L}_1 \times \mathcal{L}_2, \sqsubseteq, \sqcap, \sqcup, \top, \bot \rangle$ where:

Product Lattices

- Assume (complete) lattices:
  - $L_1 = \langle \mathcal{L}_1, \sqsubseteq_1, \sqcap_1, \sqcup_1, \top_1, \bot_1 \rangle$
  - $L_2 = \langle \mathcal{L}_2, \sqsubseteq_2, \sqcap_2, \sqcup_2, \top_2, \bot_2 \rangle$
- Let $L_1 \times L_2 = \langle \mathcal{L}_1 \times \mathcal{L}_2, \sqsubseteq, \sqcap, \sqcup, \top, \bot \rangle$ where:
  - $\langle a, b \rangle \sqsubseteq \langle a', b' \rangle$ iff $a \sqsubseteq_1 a'$ and $b \sqsubseteq_2 b'$
  - $\langle a, b \rangle \sqcap \langle a', b' \rangle = \langle a \sqcap_1 a', b \sqcap_2 b' \rangle$
  - $\langle a, b \rangle \sqcup \langle a', b' \rangle = \langle a \sqcup_1 a', b \sqcup_2 b' \rangle$
  - $\top = \langle \top_1, \top_2 \rangle$
  - $\bot = \langle \bot_1, \bot_2 \rangle$

Point-wise products of (complete) lattices are again (complete) lattices

Summary

- Complete lattices are formal basis for many program analyses
- Complete lattice $L = \langle \mathcal{L}, \sqsubseteq, \sqcap, \sqcup, \top, \bot \rangle$
  - $\mathcal{L}$: Carrier set
  - $(\sqsubseteq)$: Partial order
  - $(\sqcap)$: Join operation: find least upper lower bound
  - $(\sqcup)$: Meet operation: find greatest lower bound (not usually necessary)
  - $\top$: Top-most element of complete lattice
  - $\bot$: Bottom-most element of complete lattice
- **Product Lattices**: $L_1 \times L_2$ forms a lattice if L_1 and L_2 are lattices