# Ordnungsrelationen auf Mengen

- Eine (partielle) **Ordnungsrelation** oder kurz **Ordnung** $O$ auf einer Menge $M$ ist eine Relation, die reflexiv, antisymmetrisch und transitiv ist.

  *Beispiel:* $M = \{ 1 , 2 , 3 \}$,
  $O = \{ ( 1 , 1 ) , ( 2 , 2 ) , ( 3 , 3 ) , ( 1 , 2 ) , ( 1 , 3 ) \}$

  (Das linke Element des geordneten Paares wird als das kleinere, das rechte Element als das grössere bezeichnet.)

- Eine Ordnungsrelation $O$ auf einer Menge $M$ ist **total (oder linear)**, wenn zusätzlich gilt:

  $$\forall\, x, y \in M : ( x , y ) \in O \vee ( y , x ) \in O$$

  *Beispiel:* $M = \{ 1 , 2 \}$, $O = \{ ( 1 , 1 ) , ( 2 , 2 ) , ( 1 , 2 ) \}$

  (Wenn eine Ordnung O auf einer Menge M nicht total ist, dann gibt es Elemente x, y aus M, sodass das Paar (x, y) nicht in O ist; man sagt: x und y sind nicht vergleichbar.)

- Bekannte Beispiele von Ordnungsrelationen:

  $\subseteq$ auf Mengen (partielle Ordnung)

  $\leq$ auf ganzen oder reellen Zahlen (totale Ordnung)

- $(X, \leq)$ ist eine **partiell geordnete Menge** wenn $\leq$ eine Ordnungsrelation auf $X$ ist.

# Beispiel einer Ordnungsrelation

Sei $X$ die Menge aller syntaktisch korrekten Pascal Programme, die einen Eingabewert lesen und entweder in eine Endlosschleife gehen oder einen Wert ausgeben. Die Relation $\sqsubseteq \subseteq X \times X$ definieren wir als:

$P \sqsubseteq Q$ genau dann, wenn das Programm $Q$ für alle Eingabewerte terminiert, für die das Programm $P$ terminiert, und in diesem Fall auch dasselbe Ergebnis ausgibt wie $P$.

Ist $\sqsubseteq$ eine Ordnungsrelation?

# Spezielle Elemente von Ordnungen

- $s \in M$ heisst **kleinstes Element**. wenn $s$ kleiner als **alle** anderen Elemente ist.

$$\forall\, x \in M : (s, x) \in O$$

- $s \in M$ heisst **grösstes Element**, wenn $s$ grösser als **alle** anderen Elemente ist.

$$\forall\, x \in M : (x, s) \in O$$

- $s \in M$ heisst **minimales Element**, wenn es kein Element gibt, das kleiner als $s$ ist.

$$\forall\, x \in M : (x, s) \in O \Rightarrow x = s$$

- $s \in M$ heisst **maximales Element**, wenn es kein Element gibt, das grösser als $s$ ist.

$$\forall\, x \in M : (s, x) \in O \Rightarrow x = s$$

# Spezielle Elemente von Ordnungen

- Wenn es ein kleinstes (grösstes) Element gibt, dann ist dieses Element eindeutig.
- Es kann mehr als ein minimales (maximales) Element geben.
- In einer totalen Ordnung gibt es höchstens ein minimales (maximales) Element.
- Wenn es ein kleinstes (grösstes) Element gibt, dann ist dieses Element minimal (maximal).

  Diese Aussage gilt nicht umgekehrt.

# Strikte und reflexive partielle Ordnungen

- Eine **strikte Ordnungsrelation** *O* auf einer Menge *M* ist eine Relation, die irreflexiv, asymmetrisch und transitiv ist.

  *Beispiel:* $M = \{ 1 , 2 , 3 \}$,
  $O = \{ (2 , 1) , (2 , 3) , (1 , 3) \}$

- Wenn $\leq$ eine partielle Ordnung auf der Menge *M* ist, dann ist $<$ definiert durch

  $$x < y \Leftrightarrow x \leq y \wedge x \neq y$$

  eine strikte partielle Ordnung auf *M*.

- Wenn $<$ eine strikte partielle Ordnung auf der Menge *M* ist, dann ist $\leq$ definiert durch

  $$x \leq y \Leftrightarrow x < y \vee x = y$$

  eine (reflexive) partielle Ordnung auf *M*.

# Weitere spezielle Elemente von Ordnungen

- Seien $x$ und $y$ Elemente einer partiell geordneten Menge $M$. Falls es ein Element $z$ aus $M$ gibt, so dass

  1. $z \leq x$
  2. $z \leq y$
  3. $\forall\, t \in M : t \leq x \wedge t \leq y \Rightarrow t \leq z$

  dann heisst $z$ die **grösste untere Schranke** (Infimum) von $x$ und $y$.

$$z = x \wedge y$$

- Seien $x$ und $y$ Elemente einer partiell geordneten Menge $M$. Falls es ein Element $z$ aus $M$ gibt, so dass

  1. $x \leq z$
  2. $y \leq z$
  3. $\forall\, t \in M : x \leq t \wedge y \leq t \Rightarrow z \leq t$

  dann heisst $z$ die **kleinste obere Schranke** (Supremum) von $x$ und $y$.

$$z = x \vee y$$

# Bäume als partielle Ordnungen

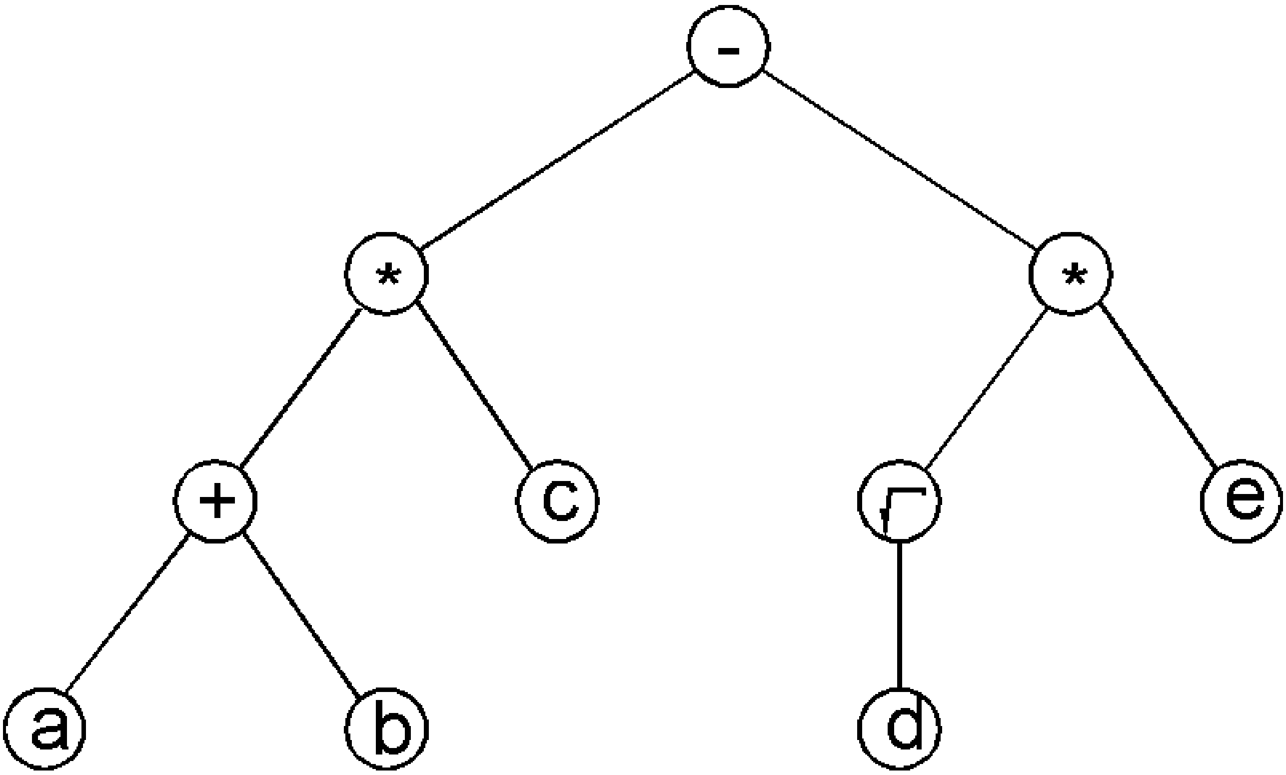
- Ein Baum ist eine partiell geordnete Menge $T$ mit einem grössten Element, so dass für jedes $x \in T$ die Menge $\{ y \in T \mid x \leq y \}$ eine endliche, total geordnete Teilmenge von $T$ ist.

- Das grösste Element heisst **Wurzel**, die minimalen Elemente heissen **Blätter**.

- Von jedem Element gibt es einen **eindeutigen Weg** zum grössten Element (Wurzel).

# Endliche, geordnete, binäre Bäume

- **Binäre Bäume** sind solche, bei denen kein Knoten mehr als zwei Kinder hat.
- **Geordnete Bäume** sind solche, bei denen die Knoten auf gleicher Ebene ebenfalls geordnet sind.
- **Endliche Bäume** haben eine endliche Anzahl Knoten.



# Anwendungen

- Filesystem
- Suchbäume
- Syntaxbäume
- Beweisbäume
- Problemlösung mit divide-and-conquer
- Verfeinerung von Lösungen
- Sportturniere mit K.O.-System
- Stammbaum
- …

# Rekursive Definition von Bäumen

- Ein einziger Knoten x ist ein binärer Baum geordnet durch $x \leq x$.

- Wenn $T_1$ und $T_2$ binäre Bäume mit $T_1 \cap T_2 = \emptyset$ sind und $x \notin T_1 \cup T_2$, dann ist die partielle Ordnung auf $\{ x \} \cup T_1 \cup T_2$, gegeben durch $y \leq x$ für alle y und $y \leq z$ wenn $y \leq z$ in $T_1$ oder $T_2$, ebenfalls ein binärer Baum.

# Traversierung von endlichen, geordneten, binären Bäumen

Eine **Traversierung** eines Baumes ist eine endliche Folge aller Knoten des Baums, die mit der Wurzel beginnt.

- **Inorder**

  Zuerst den linken Teilbaum traversieren, dann die Wurzel in die Folge schreiben, und zum Schluss den rechten Teilbaum traversieren

- **Preorder**

  Zuerst die Wurzel in die Folge schreiben, dann den linken Teilbaum traversieren und zuletzt den rechten Teilbaum traversieren.

- **Postorder**

  Zuerst den linken Teilbaum traversieren, dann den rechten Teilbaum traversieren und zuletzt die Wurzel in die Folge schreiben.

# Traversierung von endlichen, geordneten Bäumen: Tiefensuche

1. Beginne mit der Wurzel als aktuellen Knoten. Schreibe die Wurzel in die Traversierung.
2. Wähle von den noch nicht in die Traversierung geschriebenen Kindern des aktuellen Knotens das kleinste als neuen aktuellen Knoten. Schreibe den neuen aktuellen Knoten in die Traversierung.
3. Hat ein aktueller Knoten keine noch nicht in die Traversierung geschriebenen Kinder, dann wähle seinen Vater zum aktuellen Knoten.
4. Verfahre gemäss 2 und 3, bis alle Knoten des Baums aktuelle Knoten waren.

# Traversierung von endlichen, geordneten Bäumen: Breitensuche

1. Beginne mit der Wurzel als einzigem Knoten in einer Arbeitsliste. Schreibe die Wurzel in die Traversierung.
2. Ersetze der Reihe nach jeden Knoten der Arbeitsliste durch seine Kinder in der durch die Ordnung vorgegebenen Reihenfolge und schreibe diese gleichzeitig in die Traversierung.
3. Hat ein Knoten keine Kinder, so verschwindet er aus der Arbeitsliste.
4. Verfahre gemäss Regel 2 und 3, bis die Arbeitsliste leer ist.

# Traversierung von geordneten Bäumen: Limitierte Tiefensuche

- **Tiefensuche** ist zeit- und speichereffizient, beim Auftauchen von unendlichen Zweigen jedoch unvollständig
- **Breitensuche** verwendet mehr Zeit und Speicherplatz als Tiefensuche, ist jedoch auch für unendliche Bäume zu gebrauchen
- **Limitierte Tiefensuche** verbindet die Effizienz der Tiefensuche mit der Vollständigkeit der Breitensuche; ist ein Baum unendlich, schneidet man ihn auf endlicher Tiefe ab und verwendet für den nun endlichen Baum Tiefensuche; nach und nach wird die Abschneidetiefe vergrössert

# Boolesche Algebren

Eine boolesche Algebra besteht aus einer Menge $\mathbb{B}$ mit zwei ausgezeichneten Elementen $\mathbb{O}$ (Nullelement) und $\mathbb{I}$, zwei binären Operationen $\wedge$ und $\vee$ und einer unären Operation $-$.

Für $x, y, z \in \mathbb{B}$ wird vorausgesetzt:

| | |
|---|---|
| 1. Kommutativität: | $x \wedge y = y \wedge x$<br>$x \vee y = y \vee x$ |
| 2. Assoziativität: | $x \wedge (y \wedge z) = (x \wedge y) \wedge z$<br>$x \vee (y \vee z) = (x \vee y) \vee z$ |
| 3. Distributivität: | $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$<br>$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$ |
| 4. Idempotenz | $x \wedge x = x \vee x = x$ |
| 5. De Morgan | $-(x \vee y) = -x \wedge -y$<br>$-(x \wedge y) = -x \vee -y$ |
| 6. Doppelte Negation | $--x = x$ |
| 7. Invariable Elemente | $x \wedge \mathbb{O} = \mathbb{O}$<br>$x \vee \mathbb{I} = \mathbb{I}$ |
| 8. Neutrale Elemente | $x \wedge \mathbb{I} = x$<br>$x \vee \mathbb{O} = x$ |
| 9. Komplementarität | $x \wedge -x = \mathbb{O}$<br>$x \vee -x = \mathbb{I}$ |
| Weiterhin gilt: | $\mathbb{O} \neq \mathbb{I}$, $\mathbb{O} = -\mathbb{I}$ |

# Boolesche Algebren als partielle Ordnungen

- Wir definieren $a \leq b$, genau dann wenn $a \wedge b = a$ ist.
- Dann gilt:
  1. $\leq$ ist eine Ordnung auf $\mathbb{B}$
  2. $a \leq b$ genau dann, wenn $a \vee b = b$
  3. $a \leq b$ genau dann, wenn $-b \leq -a$
  4. Unter der partiellen Ordnung $\leq$ sind $a \wedge b$ bzw. $a \vee b$ die grösste untere Schranke bzw. die kleinste obere Schranke von a und b

# Beispiele boolescher Algebren 1

- Aussagenlogik:

  $\mathbb{B} = \{ \top, \bot \}$, $\mathbb{O} = \bot$, $\mathbb{I} = \top$, $\vee = \vee$, $\wedge = \wedge$, $- = \neg$

- Mengenlehre:

  Sei X eine nichtleere Menge.
  $\mathbb{B} = \mathcal{P}(X)$, $\mathbb{O} = \emptyset$, $\mathbb{I} = X$, $\vee = \cup$, $\wedge = \cap$, $- = {}^c$ (${}^c$ steht dabei für das Komplement)

# Beispiele boolescher Algebren 2

Die Menge aller Wahrheitsfunktionen auf einer Menge $p = \{ p_1, p_2, \ldots , p_n \}$ von aussagenlogischen Variablen, mit $\vee = \vee$, $\wedge = \wedge$, $- = \neg$ bildet ebenfalls eine boolesche Algebra. $\mathbb{O}$ ist die Wahrheitsfunktion, die immer $\perp$ zurückgibt, $\mathbb{1}$ die Wahrheitsfunktion, die immer $\top$ zurückgibt.

Für $P = \{ p, q \}$ z.B. gibt es 16 mögliche Wahrheitsfunktionen:

| p | q | $\mathbb{O}$ | p **nor** q | $\neg p \wedge q$ | $p \wedge \neg q$ | $p \wedge q$ | $\neg p$ | $\neg q$ | $p \Leftrightarrow \neg q$ |
|---|---|---|---|---|---|---|---|---|---|
| $\top$ | $\top$ | $\perp$ | $\perp$ | $\perp$ | $\perp$ | $\top$ | $\perp$ | $\perp$ | $\perp$ |
| $\top$ | $\perp$ | $\perp$ | $\perp$ | $\perp$ | $\top$ | $\perp$ | $\perp$ | $\top$ | $\top$ |
| $\perp$ | $\top$ | $\perp$ | $\perp$ | $\top$ | $\perp$ | $\perp$ | $\top$ | $\perp$ | $\top$ |
| $\perp$ | $\perp$ | $\perp$ | $\top$ | $\perp$ | $\perp$ | $\perp$ | $\top$ | $\top$ | $\perp$ |

| p | q | $p \Leftrightarrow q$ | q | p | p **nand** q | $\neg p \vee q$ | $p \vee \neg q$ | $p \vee q$ | $\mathbb{1}$ |
|---|---|---|---|---|---|---|---|---|---|
| $\top$ | $\top$ | $\top$ | $\top$ | $\top$ | $\perp$ | $\top$ | $\top$ | $\top$ | $\top$ |
| $\top$ | $\perp$ | $\perp$ | $\perp$ | $\top$ | $\top$ | $\perp$ | $\top$ | $\top$ | $\top$ |
| $\perp$ | $\top$ | $\perp$ | $\top$ | $\perp$ | $\top$ | $\top$ | $\perp$ | $\top$ | $\top$ |
| $\perp$ | $\perp$ | $\top$ | $\perp$ | $\perp$ | $\top$ | $\top$ | $\top$ | $\perp$ | $\top$ |

# Verbände

- Gegeben sei eine partielle Ordnung $\leq$ auf einer Menge X.

  Falls für beliebige x und y eine grösste untere Schranke $x \wedge y$ und eine kleinste obere Schranke $x \vee y$ existiert, dann heisst $( X , \leq )$ ein **Verband**.

- Jede boolesche Algebra ist ein Verband, aber nicht jeder Verband ist eine boolesche Algebra.

  Beispiel:

  $$X = \mathbb{N} \text{ mit}$$

  $x \vee y$ = kleinstes gemeinsames Vielfaches von x und y
  $x \wedge y$ = grösster gemeinsamer Faktor von x und y
  $x \leq y$ wenn $y \bmod x = 0$

  ist ein Verband, aber keine boolesche Algebra, weil es kein grösstes Element gibt.

# Schranken für beliebige Teilmengen geordneter Mengen

- Verallgemeinerung der Begriffe grösste untere Schranke und kleinste obere Schranke.

- Gegeben seien eine partiell geordnete Menge ( X, $\leq$ ) und eine Menge A mit $A \subseteq X$ und $A \neq \emptyset$. Dann ist $x \in X$ die **kleinste obere Schranke von A**, man schreibt sup(a) oder

  $x = \bigvee A$, wenn

  1. $y \in A : y \leq x$ (d.h. x ist eine obere Schranke)
  2. $\forall\ t \in X : (\ \forall\ y \in A : y \leq t\ ) \Rightarrow x \leq t$ (d.h. x ist kleiner als alle anderen oberen Schranken)

  und $z \in X$ die **grösste untere Schranke von A**, man

  schreibt inf(A) oder $z = \bigwedge A$, wenn

  1. $\forall\ y \in A : z \leq y$
  2. $\forall\ t \in X : (\ \forall\ y \in A : t \leq y\ ) \Rightarrow t \leq z$