Logik und diskrete Strukturen
(Mathematik für Informatiker II a)

Bonn, Sommersemester 2004

Vorläufiges Skript zur Vorlesung

Peter Koepke

Vorwort

Die *Mathematik für Informatiker IIa (Logik und diskrete Strukturen)* baut auf der Vorlesung *Mathematik für Informatiker Ia (Lineare Algebra)* auf. Sie ist ähnlich organisiert wie die *Lineare Algebra*. Die Vorlesung wurde folgendermaßen angekündigt:

Mathematik für Informatiker IIa
(Logik und diskrete Strukturen)

Sommersemester 2004

Vorlesung: Prof. Dr. Peter Koepke, Dr. Benedikt Löwe
Montags, 14 - 16 Uhr, HS D, Römerstrasse
Übungen: Dr. Bernhard Irrgang

Die Übungen finden in Form von Übungsgruppen statt. Die erfolgreiche Teilnahme an den Übungen ist Voraussetzung für die Zulassung zur Klausur. An den Übungen hat erfolgreich teilgenommen, wer jeweils mehr als die Hälfte der in den Übungsblättern und Präsenzübungen möglichen Punkte erreicht.

Diplomstudium Informatik: Vorlesung 2 Semesterwochenstunden (SWS) mit Übungen 2 SWS (Modul im Sinne der Diplomprüfungsordnung (DPO)).
Abschlussprüfung: Klausur Juli 2004, Note und (4) Leistungspunkte sind Teil der Vordiplomprüfung. Voraussetzung zur Zulassung zur Abschlussprüfung: aktive und erfolgreiche Teilnahme an den Übungen.

Übungsbetrieb: 2-stündige Übungsgruppen unter Anleitung von Tutoren; wöchentliche Hausaufgaben, Ausgabe montags, Abgabe bis spätestens montags vor der Vorlesung, wöchentliche Anwesenheitsaufgaben (Test) in den Übungsgruppen. Hausaufgaben können allein oder in Zweiergruppen eingereicht werden
Erfolgreiche Übungsteilnahme: jeweils 50% der möglichen Punkte in den Hausaufgaben und in den Anwesenheitsaufgaben.

Termine:
19. 04. 2004: 1. Übungsblatt, Eintragung für Übungsgruppen
1. Vorlesungswoche: Festlegung der Übungsgruppen
2. Vorlesungswoche: Beginn der Übungsgruppen, Beginn der Präsenzübungen
Juli 2004: Anmeldung zur Abschlussklausur
voraussichtlich 21. 07. 2004: Abschlussklausur
Semesterferien: Wiederholungsklausur

Die Logik formalisiert die mathematischen Methoden, die in der Vorlesung Mathematik für Informatiker Ia informell eingeführt wurden:
Mathematische Aussagen
Definitionen
Sätze
Schlussfolgerungen
Beweise.

Mathematische Aussagen beziehen sich auf Strukturen, insbesondere eine Schar immer wieder kehrender Grundstrukturen:
Zahlsysteme, Körper, Vektorräume
Relationen

Graphen
Boolesche Algebren.

Wir betrachten hier insbesondere diskrete Strukturen, bei denen kombinatorische und algorithmische Fragen im Vordergrund stehen. Die Methoden der Logik lassen sich in Analogie zu Axiomensystemen und Strukturen auch auf Programmiersprachen und Algorithmen anwenden. Wir betrachten unter diesem Gesichtspunkt auch:
Logisches Programmieren.

Literatur: Uwe Schöning, Logik für Informatiker, 5. Auflage, Spektrum Akademischer Verlag, 2000
Martin Aigner, Diskrete Mathematik, 4. Auflage, Vieweg, 2001

Kapitel 1
Prädikatenlogische Schreibweisen

In der Vorlesung *Mathematik für Informatiker Ia (Lineare Algebra)* wurde besonderer Wert auf *logisches Vorgehen*, d.h. auf vollständige *sprachliche* Formulierungen und Argumente gelegt. Es wurde eine ,,Sprache der Mathematik" eingeführt, in der in eingeschränkter und kontrollierter Weise mathematische Aussagen gebildet werden konnten. Der Hauptbegriff der linearen Algebra wurde beispielsweise folgendermaßen erfasst:

... . Dann ist $V = (V, +, \cdot)$ ein $\mathbb{K}$-**Vektorraum** oder ein **Vektorraum über** $\mathbb{K}$, wenn die folgenden Axiome gelten:

a) Für $x, y, z \in V$ gilt $(x + y) + z = x + (y + z)$ (Assoziativgesetz).

b) Für $x, y \in V$ gilt $x + y = y + x$ (Kommutativgesetz).

c) Es gibt ein $0 \in V$, so dass für $x \in V$ gilt $x + 0 = x$ (Existenz eines **Nullvektors**).

d) Für $x \in V$ gibt es ein $-x \in V$, so dass $x + (-x) = 0$ (Existenz additiver Inverser).

e) Für $\lambda, \mu \in \mathbb{K}$ und $x \in V$ gilt $\lambda(\mu x) = (\lambda\mu)x$ (Assoziativgesetz).

f) Für $x \in V$ gilt $1x = x$ (Neutralität der 1).

g) Für $\lambda \in \mathbb{K}$ und $x, y \in M$ gilt $\lambda(x + y) = \lambda x + \lambda y$ (1. Distributivgesetz).

h) Für $\lambda, \mu \in \mathbb{K}$ und $x \in M$ gilt $(\lambda + \mu)x = \lambda x + \mu x$ (2. Distributivgesetz).

Dieser mathematische Text benutzt nur wenige sprachliche Figuren: ,,Für ...", ,,gibt es ..." usw. Es bietet sich von daher an, *abkürzende Notationen* für diese Sprach-Figuren einzuführen.

In der *Linearen Algebra* hatten wir die Bildung von mathematischen *Ausdrücken* diskutiert. Die einfachsten Ausdrücke sind die *atomaren Ausdrücke*, die *relationale Aussagen* über *Terme* machen. Wir bilden zunächst wie gewohnt aus Variablen, Konstanten und Funktionssymbolen *Terme*:

$$c,\ a + b,\ a^n,\ \log(x) \text{ usw.}$$

Wenn $t_0, t_1, ...$ Terme und $=, <, R(.,.)$ Symbole für *Relationen* sind, so lassen sich hieraus *atomare Ausdrücke*

$$t_0 = t_1\,,\ t_0 < t_1\,,\ R(t_0, t_1, ...) \text{ usw.}$$

bilden. Man erhält komplexe Ausdrücke mit Hilfe folgender *Regeln*:

Aussagenlogische Verknüpfungen:

- Konjunktion: ,,A und B", ,,es gelten A und B", oder manchmal auch nur ,,A, B".
- Disjunktion: ,,A oder B", ,,es gilt A oder B".
- Negation: ,,nicht A", ,,A gilt nicht".

Quantorenlogische Verknüpfungen:

- Existenz: ,,es gibt ein x mit A", ,,es existiert ein x mit A", ,,es gibt ein x, so dass A", ,,es gibt ein x mit A, so dass B"; ,,es gibt $x_0, ..., x_{k-1}$ mit A".
- Allquantor: ,,für alle x gilt A", ,,für alle x ist A", ,,für x gilt A"; ,,für alle $x_0, ..., x_{k-1}$ gilt A".
- Bedingter Allquantor: ,,für alle x mit A gilt B", ,,für alle x mit A ist B", ,,für x mit A gilt B"; ,,für alle $x_0, ..., x_{k-1}$ mit A gilt B".
- Eindeutige Existenz: ,,es gibt genau ein x mit A".

Wir führen die Abkürzungen $\wedge, \vee, \neg, \exists, \forall$ anstelle von ,,und", ,,oder", ,,nicht", ,,es existiert" und ,,für alle" ein:

- Konjunktion: $A \wedge B$ für ,,A und B",
- Disjunktion: $A \vee B$ für ,,A oder B",
- Negation: $\neg A$ für ,,nicht A",
- Existenzielle Quantifizierung: $\exists x\, A$ für ,,es gibt ein x mit A",
- Universelle Quantifizierung: $\forall x\, A$ für ,,für alle x gilt A".

Die Vektorraum-Definition kann mit diesen Abkürzungen folgendermaßen geschrieben werden:

a) $\forall x, y, z \in V\colon (x + y) + z = x + (y + z)$ (Assoziativgesetz).

b) $\forall x, y \in V\colon\ x + y = y + x$ (Kommutativgesetz).

c) $\forall x \in V\colon\ x + 0 = x$ (Nullvektor).

d) $\forall x \in V\, \exists y \in V\colon x + y = 0$ (Existenz additiver Inverser).

e) $\forall \lambda, \mu \in \mathbb{K}\, \forall x \in V\colon\ \lambda \cdot (\mu \cdot x) = (\lambda \cdot \mu) \cdot x$ (Assoziativgesetz).

f) $\forall x \in V\colon\ 1 \cdot x = x$ (Neutralität der 1).

g) $\forall \lambda \in \mathbb{K}\, \forall x, y \in M\colon\ \lambda \cdot (x + y) = \lambda \cdot x + \lambda \cdot y$ (1. Distributivgesetz).

h) $\forall \lambda, \mu \in \mathbb{K}\, \forall x \in M\colon\ (\lambda + \mu) \cdot x = \lambda \cdot x + \mu \cdot x$ (2. Distributivgesetz).

Kapitel 2
Relationen

Wir erinnern an die Definition einer *Relation* aus der Linearen Algebra:

Definition 2.1. *a) R ist eine n-stellige **Relation**, wenn R eine Menge von n-Tupeln ist. Statt $(x_0, ..., x_{n-1}) \in R$ schreiben wir auch $R(x_0, ..., x_{n-1})$. Im Fall $n=2$ schreiben wir statt $R(x,y)$ auch xRy (Infix-Notation).*

b) R ist eine Relation auf A und B, wenn $R \subseteq A \times B$. R ist eine 2-stellige Relation auf A, wenn $R \subseteq A \times A$.

Relationen lassen sich 2-dimensional graphisch darstellen; eine Relation kann mit ihrem Graphen identifiziert werden. Wir definieren wichtige Eigenschaften von Relationen mit Hilfe prädikatenlogischer Schreibweisen.

Definition 2.2. *a) R ist eine **Funktion** von A nach B, $R\colon A \to B$, wenn $\forall a \in A \exists b \in B (aRb \wedge \forall b' \in B (aRb' \to b = b'))$.*

*b) R ist eine **injektive Funktion** von A nach B, wenn $(R\colon A \to B \wedge \forall a, a' \in A \forall b, b' \in B ((aRb \wedge a'Rb' \wedge a \neq a') \to b \neq b'))$.*

*c) R ist eine **surjektive Funktion** von A auf B, wenn $(R\colon A \to B \wedge \forall b \in B \exists a \in A\, aRb)$.*

*d) R ist eine **Bijektion** zwischen A und B, $R\colon A \leftrightarrow B$, wenn $(R\colon A \to B \wedge R$ ist injektiv $\wedge\, R$ ist surjektiv$)$.*

Definition 2.3. *Sei R eine 2-stellige Relation auf A. Dann definiere*

*a) R ist **symmetrisch**, wenn $\forall a, b \in A (aRb \to bRa)$.*

*b) R ist **antisymmetrisch**, wenn $\forall a, b \in A ((aRb \wedge bRa) \to a = b)$.*

*c) R ist **reflexiv**, wenn $\forall a \in A\, aRa$.*

*d) R ist **transitiv**, wenn $\forall a, b, c \in A ((aRb \wedge bRc) \to aRc)$.*

*e) R ist eine **Äquivalenzrelation**, wenn R symmetrisch, reflexiv und transitiv ist.*

2.1 Äquivalenzrelationen

Definition 2.4. *Sei R eine Äquivalenzrelation auf A. Für $a \in A$ ist die Äquivalenzklasse von a bezüglich R*

$$[a] = [a]_R = \{b \mid bRa\}.$$

Satz 2.5. *Sei R eine Äquivalenzrelation auf A. Dann gilt:*

a) $\forall a, b \in A\ ([a] = [b] \vee [a] \cap [b] = \emptyset)$.

b) $A = \bigcup_{a \in A} [a]$.

Beweis. a) Betrachte $a, b \in A$.
Fall 1: $a R b$:
Behauptung: $[a] = [b]$.
Beweis: Betrachte $x \in [a]$. Dann ist $x R a$; $a R b$; $x R b$, wegen der Transitivität von R; $x \in [b]$. Also ist $[a] \subseteq [b]$.
Den Beweis der umgekehrten Inklusion notieren wir noch knapper:
$\Big[x \in [b]; x R b; b R a$, wegen der Symmetrie von R; $x R a$, wegen der Transitivität von R; $x \in [a]\Big]$ $[b] \subseteq [a]$. *qed*(Behauptung)
Dabei bezeichnet der *Rahmen*

$$[A_0; A_1; ...; A_{m-1}]$$

ein *Teilargument*, in dem ausgehend von der Annahme A_0 sukzessiv A_1, ..., A_{m-1} gezeigt werden. Nach Abschluss des Teilarguments ist die Implikation

$$A_0 \to A_{m-1}$$

bewiesen.
Fall 2: $\neg a R b$:
Behauptung: $[a] \cap [b] = \emptyset$.
Beweis: $\Big[x \in [a] \cap [b]; x R a; x R b; a R x$, wegen der Symmetrie von R; $a R b$ wegen der Transitivität von R; Widerspruch zur Fallannahme$\Big]$ $\forall x (x \in [a] \cap [b] \to 0 = 1)$; $\forall x (x \notin [a] \cap [b])$; $[a] \cap [b] = \emptyset$. *qed*(Behauptung)

b) Wegen der Reflexivität von R gilt $\forall b \in A\, b R b$. Daraus folgt $\forall b \in A\, b \in [b]$ und $\forall b \in A\, b \in \bigcup_{a \in A} [a]$. $A \subseteq \bigcup_{a \in A} [a]$. Die Gegenrichtung $A \supseteq \bigcup_{a \in A} [a]$ ist trivial. Also ist $A = \bigcup_{a \in A} [a]$. □

Beispiel 2.6. *Faktorisieren nach Unterräumen.* Es sei V ein $\mathbb{K}$-Vektorraum und U ein Unterraum von V. Definiere eine Relation $R \subseteq V \times V$ durch

$$x R y \longleftrightarrow x - y \in U.$$

Wir zeigen, dass R eine Äquivalenzrelation auf V ist:
(1) R ist reflexiv.
Beweis: Betrachte $x \in V$. Dann ist $x - x \in U$ und $x R x$. Also gilt $\forall x \in V x R x$. *qed*
Die Struktur des Argumentes kann folgendermaßen notiert werden:
$\Big[x \in V; x - x = 0 \in U; x R x\Big]; \forall x \in V x R x$.
(2) R ist symmetrisch.
$\Big[x, y \in V, x R y; x - y \in U; y - x = -(x - y) \in U; y R x\Big]; \forall x, y \in V\, (x R y \to y R x)$.
(3) R ist transitiv.
$\Big[x, y, z \in V, x R y, y R z; x - y \in U; z - y \in U; x - z = (x - y) + (y - z) \in U; x R z\Big]$;
$\forall x, y, z \in V\, ((x R y \wedge y R z) \to x R z)$.

Definition 2.7. *Sei R eine Äquivalenzrelation auf A. Dann sei*

$$A/R = \{[a]_R \,|\, a \in A\}$$

die Menge der Äquivalenzklassen von R. A/R ist die ***Faktorisierung*** *oder der* ***Quotient*** *von A nach R.*

Beispiel 2.8. *Quotientenvektorräume.* In dem Beispiel 2.6 sind die Äquivalenzklassen von der Gestalt

$$[x]_R = x + U = \{x + u \,|\, u \in U\}.$$

Beweis: Betrachte $v \in [x]_R$. Dann ist vRx, $u = v - x \in U$, $v = x + u \in x + U$. Umgekehrt betrachte $v \in x + U$. Wähle $u \in U$ mit $v = x + u$. Dann ist $v - x = u \in U$ und vRx, $v \in [x]_R$. *qed*

Die Struktur der Umkehrung ist:
$\big[v \in x+U\colon \big[u \in U, v = x+u; v-x \in U; vRx; v \in [x]_R\big]; v \in [x]_R\big]$.

Geometrisch sind die Äquivalenzklassen *Parallelverschiebungen* $x + U$ des Untervektorraums U um den Vektor x. Nach Satz 2.5 zerfällt der Raum V in zu U parallele Untermengen.

Man kann weiter zeigen, dass die Menge A/R mit folgenden Operationen zu einem $\mathbb{K}$-Vektorraum $A/R = (A/R, \oplus, \odot)$ erweitert werden kann:

$$\begin{aligned} (x+U) \oplus (y \oplus U) &= (x+y)+U; \\ \lambda \odot (x+U) &= (\lambda \cdot x)+U. \end{aligned}$$

Beispiel 2.9. *Modulare Arithmetik.* Fixiere $n \in \mathbb{N}$, $n \neq 0$. Wir hatten die Relation $(\mathrm{mod}\, n)$ der *Kongruenz modulo n* auf den ganzen Zahlen $\mathbb{Z}$ definiert:

$$i \equiv j \,(\mathrm{mod}\, n) \text{ gdw. } n \,|\, (i-j).$$

Diese Relation ist eine Äquivalenzrelation auf $\mathbb{Z}$; der Beweis verläuft ähnlich wie bei der Bildung des Quotientenraums bei Vektorräumen. Die Äquivalenzklassen $[i]$ bezüglich dieser Relation sind von der Gestalt

$$[i] = i + n \cdot \mathbb{Z} = \{i + n \cdot k \,|\, k \in \mathbb{Z}\}.$$

Wir hatten die Äquivalenzklassen durch die *Reste* modulo n repräsentiert:

$$\mathbb{Z}_n = \{0, 1, ..., n-1\}.$$

Die natürliche Korrepondenz φ zwischen $\mathbb{Z}_n$ und $\mathbb{Z}/(\mathrm{mod}\, n)$ ist durch

$$i \longmapsto [i] = i + n \cdot \mathbb{Z}$$

gegeben. Die Korrespondenz ist mit den Operationen $\oplus_n$ und $\otimes_n$ der Arithmetik modulo n verträglich:

$$\begin{aligned} \varphi(i \oplus_n j) &= (i \oplus_n j) + n \cdot \mathbb{Z} = (i+j) + n \cdot \mathbb{Z} = (i + n \cdot \mathbb{Z}) \oplus (j + n \cdot \mathbb{Z}) = \varphi(i) \oplus \varphi(j) \\ \varphi(i \otimes_n j) &= (i \otimes_n j) + n \cdot \mathbb{Z} = (i \cdot j) + n \cdot \mathbb{Z} = (i + n \cdot \mathbb{Z}) \otimes (j + n \cdot \mathbb{Z}) = \varphi(i) \otimes \varphi(j). \end{aligned}$$

2.2 Ordnungsrelationen

Definition 2.10. *Eine 2-stellige Relation $\leqslant$ auf X ist eine* ***partielle Ordnung****, wenn $\leqslant$ transitiv, reflexiv und antisymmetrisch ist. Die Relation $\leqslant$ ist eine* ***(totale) Ordnung*** *oder* ***lineare Ordnung****, wenn $\leqslant$ außerdem* ***linear*** *ist:*

$$\forall x, y \in X\,(x \leqslant y \vee x = y \vee y \leqslant x).$$

Beispiel 2.11. a) Die *Inklusion* $\subseteq$ auf Mengen ist eine partielle Ordnung. Die Inklusion ist nicht total, denn es gibt nicht-leere, zueinander disjunkte Mengen:

$$\{0\} \not\subseteq \{1\},\ \{0\} \neq \{1\},\ \{1\} \not\subseteq \{0\}.$$

Die Antisymmetrie der Inklusion ist das fundamentale Kriterium für die Gleichheit von Mengen (*Extensionalität*), das häufig für den Beweis von Mengengleichheit benutzt wird:

$$\forall x, y\,((x \subseteq y \wedge y \subseteq x) \to x = y).$$

b) Die gewöhnlichen Ordnungen $\leqslant$ auf den Zahlbereichen $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ und $\mathbb{R}$ sind totale Ordnungen.

Zu jeder partiellen Ordnung lässt sich eine *strenge* oder *strikte* Variante definieren:

Definition 2.12. *Sei $\leqslant$ eine partielle Ordnung auf X. Definiere dann eine zweistellige Relation $<$ auf X durch:*

$$x < y \text{ gdw. } (x \leqslant y \wedge x \neq y).$$

Diese Relation ist transitiv und irreflexiv:

$$\forall x, y, z \in X\,((x < y \wedge y < z) \to x < z),$$
$$\forall x \in X\, \neg x < x.$$

Wenn zusätzlich $\leqslant$ total ist, so erfüllt $<$ die folgende Trichotomie:

$$\forall x, y \in X\,(x < y \vee x = y \vee y < x).$$

Partielle Ordnungen (in nicht-strikter und strikter Form) treten häufig auf. Wir führen einige wichtige Begriffe für partielle Ordnungen ein. Für eine partielle Ordnung $\leqslant$ auf X definiere den Ausdruck

$$\max(a, X) \leftrightarrow (a \in X \wedge \forall x \in X\colon x \leqslant a),$$

der das **Maximum** von X charakterisiert.

Satz 2.13. *Eine partielle Ordnung hat höchstens ein Maximum:*

$$\forall a, a'\,((\max(a, X) \wedge \max(a', X) \to a = a').$$

Wenn es existiert, bezeichnen wir das Maximum a von X mit

$$a = \max(X).$$

Beweis. Betrachte $a, a' \in X$ mit $\max(a, X)$ und $\max(a', X)$. Dann ist

$$\forall x \in X : x \leqslant a \quad \text{und} \quad \forall x \in X : x \leqslant a'.$$

Speziell ist $a' \leqslant a$ und $a \leqslant a'$. Wegen der Antisymmetrie von $\leqslant$ ist $a = a'$. □

Definition 2.14. *Sei $(X, \leqslant)$ eine partielle Ordnung. Definiere*

a) $\mathrm{os}(a, Y) \leftrightarrow (a \in X \wedge \forall y \in Y : y \leqslant a)$*; $\mathrm{os}(a, Y)$ besagt, dass a eine* ***obere Schranke*** *von Y ist;*

b) $\sup(a, Y) \leftrightarrow \mathrm{os}(a, Y) \wedge \forall b(\mathrm{os}(b, Y) \rightarrow b \geqslant a)$*; $\sup(a, Y)$ besagt, dass a eine* ***kleinste obere Schranke*** *oder ein* ***Supremum*** *von Y ist.*

Satz 2.15. *Wenn $\sup(a, Y)$ und $\sup(a', Y)$, dann ist $a = a'$. Wir können daher $a = \sup(Y)$ anstelle von $\sup(a, Y)$ schreiben.*

Beweis. Übung. □

Definition 2.16. *Definiere $\mathrm{us}(a, Y) \leftrightarrow \forall x \in Y\, a \leqslant x$. Dies besagt, dass a eine* ***untere Schranke*** *von Y ist. a ist ein* ***Infimum*** *von Y, wenn $\inf(a, Y) \leftrightarrow \mathrm{us}(a, Y) \wedge \forall b(\mathrm{us}(b, Y) \rightarrow b \leqslant a)$. Wie das Supremum, so ist ein Infimum eindeutig definiert, wenn es existiert. Daher schreiben wir auch $a = \inf(Y)$ anstelle von $\inf(a, Y)$*

Satz 2.17. *Angenommen, $\max(X)$ existiert. Dann ist $\max(X) = \inf(\emptyset)$.*

Beweis. $\mathrm{us}(a, \emptyset)$ ist äquivalent zu: $\forall x \in \emptyset\, a \leqslant x$. Diese Eigenschaft ist immer erfüllt. Daher

$$\inf(a, \emptyset) \leftrightarrow \forall a'\, a' \leqslant a \leftrightarrow a = \max(X).$$ □

Beispiel 2.18. (Kleine) endliche Beispiele von partiellen Ordnungen lassen sich durch **Hasse-Diagramme** angeben. Das sind Figuren, in denen die Elemente der Trägermenge durch Kanten verbunden sind. Wenn a und b durch eine Kante verbunden sind und a oberhalb von b liegt, so bedeutet das, dass $a \geqslant b$ ist. Die Relation $\leqslant$ besteht aus allen Paaren, die durch einen von unten nach oben laufenden Kantenzug verbunden werden können.

Abbildung 2.1. Hasse-Diagramm der partiellen Relation $m \,|\, n \,|\, 12$.

1. Lineare endliche partielle Ordnung.
2. So eine Raute.
3. Die nicht-distributive mit 5 Punkten.
4. Die Teiler der Zahl 24. Hier könnte man sich überlegen, dass Suprema und Infima existieren. Die Suprema sind die kleinsten gemeinsamen Vielfachen, die Infima sind die größten gemeinsamen Teiler.

4. Das ganze auch für $\mathbb{N}$. Dabei geht alles entsprechend, allerdings gibt es kein maximales Element.

Existenz von Suprema und Infima in partiell geordneten Mengen. Wir können verschiedene Beispiele angeben.

Wir interessieren uns besonders für Suprema und Infima endlicher Mengen.

Definition 2.19. *Sei $(X, \leqslant)$ eine partielle Ordnung. $a \sqcup b = \sup(\{a, b\})$ ist die **Vereinigung** von a und b. $a \sqcap b = \inf(\{a, b\})$ ist der **Schnitt** von a und b.*

Satz 2.20. *Wenn $\subseteq$ die partielle Ordnung der **Inklusion** auf Mengen ist, so gilt:*

a) $\forall a, b: a \sqcup b = a \cup b$.

b) $\forall a, b: a \sqcap b = a \cap b$.

Beweis. a) Betrachte Mengen a und b.
Es gilt $a \subseteq a \cup b$ und $b \subseteq a \cup b$. Daher ist $\mathrm{os}(a \cup b, \{a, b\})$. Betrachte ein c mit $\mathrm{os}(c, \{a, b\})$. Dann ist $a \subseteq c$ und $b \subseteq c$. Zusammen ist $a \cup b \subseteq c$. Also ist $\forall c\,(\mathrm{os}(c, \{a, b\}) \to a \cup b \subseteq c)$. Damit ist

$$a \cup b = \sup(\{a, b\}) = a \sqcup b.$$

b) lässt sich analog zeigen. □

Definition 2.21. *Eine partiell geordnete Menge $(X, \leqslant)$ ist ein **Verband**, wenn die Suprema und Infima aller endlichen Teilmengen existieren.*

Satz 2.22. *Sei $(X, \leqslant)$ ein Verband. Dann besitzt $(X, \leqslant)$ ein Maximum und ein Minimum, nämlich* $\inf(\emptyset)$ *und* $\sup(\emptyset)$. *Wir bezeichnen das Maximum und das Minimum mit $\top$ und $\bot$ (**Top** und **Bottom**).*

Folgende Gesetze gelten in Verbänden:

Satz 2.23. *Sei X ein Verband. Dann gilt **in** X:*

a) $\forall x, y: x \leqslant x \sqcup y$ *und* $\forall x, y: x \geqslant x \sqcap y$.

b) $\forall x: x \sqcup x = x$ *und* $\forall x: x \sqcap x = x$ *(Idempotenz).*

c) $\forall x, y: x \sqcup y = y \sqcup x$ *und* $\forall x, y: x \sqcap y = y \sqcap x$ *(Kommutativität).*

d) $\forall x, y, z: x \sqcup (y \sqcup z) = (x \sqcup y) \sqcup z$ *und* $\forall x, y, z: x \sqcap (y \sqcap z) = (x \sqcap y) \sqcap z$ *(Assoziativität).*

e) $\forall x, y: x \sqcap (x \sqcup y) = x$ *und* $\forall x, y: x \sqcup (x \sqcap y) = x$ *(Absorption).*

f) $\forall x: \bot \sqcup x = x$ *und* $\forall x: \bot \sqcap x = \bot$ *(Minimum).*

g) $\forall x\colon \top \sqcap x = x$ *und* $\forall x\colon \top \sqcup x = \top$ *(Maximum).*

Beweis. a) Betrachte $x, y \in X$. $\mathrm{os}(x \sqcup y, \{x, y\})$ und $x \sqcup y \geqslant x$.
b) Betrachte $x \in X$. $x \geqslant x$ und $\mathrm{os}(x, \{x, x\})$. Betrachte ein $y \in X$ mit $\mathrm{os}(y, \{x, x\})$. Dann ist $y \geqslant x$. Also ist $\forall y \in X\,(\mathrm{os}(y, \{x, x\}) \to y \geqslant x)$. Zusammen ist

$$(\mathrm{os}(x, \{x, x\}) \wedge \forall y \in X\,(\mathrm{os}(y, \{x, x\}) \to y \geqslant x)),$$

d.h. $x = \sup(\{x, x\}) = x \sqcup x$.
d) Betrachte $x, y, z \in X$. Dann gilt

$$\begin{gathered} x \sqcup (y \sqcup z) \geqslant x \\ x \sqcup (y \sqcup z) \geqslant y \sqcup z \geqslant y \\ x \sqcup (y \sqcup z) \geqslant x \sqcup y \\ x \sqcup (y \sqcup z) \geqslant y \sqcup z \geqslant z \\ x \sqcup (y \sqcup z) \geqslant (x \sqcup y) \sqcup z \end{gathered}$$

Umgekehrt:

$$\begin{gathered} x \leqslant x \sqcup y \leqslant (x \sqcup y) \sqcup z \\ y \leqslant x \sqcup y \leqslant (x \sqcup y) \sqcup z \\ z \leqslant (x \sqcup y) \sqcup z \\ y \sqcup z \leqslant (x \sqcup y) \sqcup z \\ x \sqcup (y \sqcup z) \leqslant (x \sqcup y) \sqcup z \end{gathered}$$

Wegen der Antisymmetrie von $\leqslant$ ist dann

$$x \sqcup (y \sqcup z) = (x \sqcup y) \sqcup z.$$

e) Betrachte $x, y \in X$. Wir zeigen die Gleichheit durch zwei Ungleichungen:
Beh: $x \sqcap (x \sqcup y) \leqslant x$. Dies gilt nach a).
Beh: $x \sqcap (x \sqcup y) \geqslant x$. $x \geqslant x$ und nach a) $x \sqcup y \geqslant x$. Also ist $\mathrm{us}(x, \{x, x \sqcup y\})$ und $x \leqslant \inf(x, x \sqcup y) = x \sqcap (x \sqcup y)$.
Da $\leqslant$ antisymmetrisch ist, ist $x \sqcap (x \sqcup y) = x$. □

Beim Rechnen mit *zwei* Rechenarten sind Distributivgesetze interessant. Für allgemeine Verbände gilt:

Satz 2.24. *Sei X ein Verband.*

a) $\forall x, y, z\colon (y \geqslant z \to x \sqcap y \geqslant x \sqcap z)$

b) $\forall x, y, z\colon x \sqcap (y \sqcup z) \geqslant (x \sqcap y) \sqcup (x \sqcap z)$

c) $\forall x, y, z\colon x \sqcup (y \sqcap z) \leqslant (x \sqcup y) \sqcap (x \sqcup z)$.

Beweis. a) Betrachte x, y, $z \in X$ mit $y \geqslant z$. Dann ist $x \sqcap z \leqslant x$. $x \sqcap z \leqslant z \leqslant y$. $\mathrm{us}(x \sqcap z, \{x, y\})$. Also $x \sqcap z \leqslant \inf(\{x, y\}) = x \sqcap y$.
b) Betrachte $x, y, z \in X$. Dann ist $x \geqslant x \sqcap y$, $x \geqslant x \sqcap z$ und $x \geqslant (x \sqcap y) \sqcup (x \sqcap z)$. Weiter ist $y \sqcup z \geqslant y \geqslant x \sqcap y$, $y \sqcup z \geqslant z \geqslant x \sqcap z$ und $y \sqcup z \geqslant (x \sqcap y) \sqcup (x \sqcap z)$. Also ist $(x \sqcap y) \sqcup (x \sqcap z)$ untere Schranke von $\{x, y \sqcup z\}$ und $x \sqcap (y \sqcup z) \geqslant (x \sqcap y) \sqcup (x \sqcap z)$.
c) lässt sich mit einem zum Beweis von b) „dualen" Argument zeigen, bei dem $\sqcap$ und $\sqcup$ bzw. $\leqslant$ und $\geqslant$ vertauscht sind. □

Definition 2.25. *Ein Verband $(X, \leqslant)$ ist ein **distributiver Verband**, wenn in ihm die **Distributivgesetze** gelten:*

$$\forall x, y, z: x \sqcap (y \sqcup z) = (x \sqcap y) \sqcup (x \sqcap z)$$
$$\forall x, y, z: x \sqcup (y \sqcap z) = (x \sqcup y) \sqcap (x \sqcup z).$$

Beispiel 2.26. Hasse-Diagramme von distributivem Verband mit 5 Elementen, einmal distributiv, einmal nicht-distributiv.

Beispiel 2.27. Der Teilbarkeitsverband (das ist der Verband mit kleinsten und größten gemeinsamen Teilern) ist distributiv: Übung?

2.3 Boolesche Algebren

Wenn wir den Verband der Teilmengen einer Menge Z betrachten, so gibt es dort noch eine weitere wichtige Operation, nämlich die Komplementbildung:

$$A \mapsto -A = \{x \in Z \mid x \notin A\} = Z \setminus A.$$

Das Komplement ist dadurch gekennzeichnet, dass $A \cap (-A) = \emptyset$ und $A \cup (-A) = Z$. Wir können dieses Phänomen abstrakt in beliebigen distributiven Verbänden studieren.

Definition 2.28. *Sei $(X, \leqslant)$ ein distributiver Verband. Elemente x, $x' \in X$ heißen **komplementär**, wenn $x \sqcup x' = \top$ und $x \sqcap x' = \bot$.*

Satz 2.29. *Sei $(X, \leqslant)$ ein distributiver Verband und $x, x', x'' \in X$. Wenn x und x' als auch x und x'' komplemetär sind, so ist $x' = x''$. Dieses **Komplement** x' von x wird auch mit $-x$ bezeichnet.*

Beweis. Betrachte x, x', x'' mit den genannten Eigenschaften.

$$\begin{aligned} x' &= x' \sqcap \top \\ &= x' \sqcap (x \sqcup x'') \\ &= (x' \sqcap x) \sqcup (x' \sqcap x'') \\ &= \bot \sqcup (x' \sqcap x'') \\ &= x' \sqcap x'' \end{aligned}$$

Daraus folgt $x' \leqslant x''$. Genauso zeigt man die umgekehrte Ungleichung $x'' \leqslant x'$. Zusammen gilt dann $x' = x''$. □

Definition 2.30. *Eine **Boolesche Algebra** ist ein distributiver Verband $(X, \leqslant)$ mit der Eigenschaft:*

$$\forall x \in X \, \exists x' \in X \, (x \text{ und } x' \text{ sind komplementär}).$$

In diesem Verband können wir die Operationen $\sqcup$, $\sqcap$, $-$ betrachten. Wir bevorzugen allerdings folgende algebraische Definition von Booleschen Algebren.

Definition 2.31. *Eine* ***Boolesche Algebra*** *ist eine Struktur* $(B, +, \cdot, -, 0, 1)$ *mit zwei zweistelligen Funktionen* $+, \cdot$, *einer einstelligen Funktion* $-$, *und zwei Konstanten* 0 *und* 1, *die die folgenden Axiome erfüllt:*

a) $\forall x, y, z: x + (y + z) = (x + y) + z$, $\forall x, y, z: x \cdot (y \cdot z) = (x \cdot y) \cdot z$;

b) $\forall x, y: x + y = y + x$, $\forall x, y: x \cdot y = y \cdot x$;

c) $\forall x: 0 + x = x$, $\forall x: 1 \cdot x = x$;

d) $\forall x, y, z: x \cdot (y + z) = (x \cdot y) + (x \cdot z)$, $\forall x, y, z: x + (y \cdot z) = (x + y) \cdot (x + z)$;

e) $\forall x: x + (-x) = 1$, $\forall x: x \cdot (-x) = 0$;

f) $-0 = 1$, $-1 = 0$, $0 \neq 1$;

g) $\forall x: x \cdot 0 = 0$, $\forall x: x + 1 = 1$;

h) $\forall x: -(-x) = x$;

i) $\forall x: x \cdot x = x$, $\forall x: x + x = x$;

j) $\forall x, y: -(x \cdot y) = (-x) + (-y)$, $\forall x, y: -(x + y) = (-x) \cdot (-y)$.

Die Axiome j) sind die **de Morgan*schen Gesetze***.

Beispiel 2.32. Das einfachste Beispiel einer Booleschen Algebra besteht nur aus den Elementen 0 und 1: $B = \{0, 1\}$. Die Verknüpfungen $+, \cdot$ und $-$ müssen auf Grund der Axiome folgendermaßen definiert werden:

$+$	0	1
0	0	1
1	1	1

,

$\cdot$	0	1
0	0	0
1	0	1

,

$-$	
0	1
1	0

.

Diese Algebra ist *isomorph* zur Algebra der *Wahrheitswerte*

oder	**W**	**F**
W	**W**	**W**
F	**W**	**F**

,

und	**W**	**F**
W	**W**	**F**
F	**F**	**F**

,

nicht	
W	**F**
F	**W**

,

die wir im letzten Semester kennen gelernt hatten.

Beispiel 2.33. *Potenzmengen*: Sei $X \neq \emptyset$ eine nicht-leere Menge. Definiere die **Potenzmenge** von X als die Menge aller Teilmengen von X:

$$\mathrm{Pot}(X) = \{Y \mid Y \subseteq X\}.$$

Dann ist $\mathrm{Pot}(X)$ mit den Operationen $\cup, \cap$ und Komplementbildung

$$-Y = X \setminus Y = \{x \in X \mid x \notin Y\}$$

eine Boolesche Algebra.

Satz 2.34. *Sei* X *eine endliche Menge mit* n *Elementen,* $n \in \mathbb{N}$. *Dann besitzt* $\mathrm{Pot}(X)$ *genau* 2^n *Elemente.*

Beweis. Durch vollständige Induktion über $n \in \mathbb{N}$.
Induktionsanfang $n = 0$: Dann ist X die *leere Menge*, $X = \emptyset$. Die einzige Teilmenge von $\emptyset$ ist die leere Menge selbst:

$$\mathrm{Pot}(\emptyset) = \{\emptyset\}.$$

Damit hat $\mathrm{Pot}(\emptyset)$ genau $1 = 2^0$ Elemente.
Induktionsschritt: die Behauptung gelte für n. Betrachte eine Menge X mit $n+1$ Elementen. Wähle ein $a \in X$. Die Menge $X \setminus \{a\}$ hat n Elemente. Dann ist

$$\begin{aligned} \mathrm{Pot}(X) &= \{Y \subseteq X \,|\, a \in Y\} \cup \{Y \subseteq X \,|\, a \notin Y\} \\ &= \{Z \cup \{a\} | Z \in \mathrm{Pot}(X \setminus \{a\})\} \cup \mathrm{Pot}(X \setminus \{a\}). \end{aligned}$$

Nach Induktionsvoraussetzung hat $\mathrm{Pot}(X \setminus \{a\})$ genau 2^n Elemente. Die beiden „Summanden" der Vereinigung haben daher jeweils 2^n Elemente. Zusammen hat die Vereinigung und damit die Potenzmenge von X $2^n + 2^n = 2^{n+1}$ Elemente.
Der Satz folgt mit dem Prinzip der vollständigen Induktion. □

Satz 2.35. *Sei $(B, +, \cdot, -, 0, 1)$ eine Boolesche Algebra. Dann gilt in B:*

a) $\forall x, y\colon (x + y = 0 \to (x = 0 \wedge y = 0))$.

b) $\forall x, y\colon (x \cdot (-y) = 0 \leftrightarrow x \cdot y = x)$.

Beweis. a) Betrachte $x, y \in B$ mit $x + y = 0$. Dann ist

$$x = x + 0 = x + (x + y) = (x + x) + y = x + y = 0.$$

Ebenso ist $y = 0$.
b) Betrachte $x, y \in B$. Angenommen $x \cdot (-y) = 0$. Dann

$$\begin{aligned} x \cdot y &= (x \cdot y) + 0 \\ &= (x \cdot y) + (x \cdot (-y)) \\ &= x \cdot (y + (-y)) \\ &= x \cdot 1 \\ &= x. \end{aligned}$$

Andererseits sei $x \cdot y = x$. Dann

$$\begin{aligned} x \cdot (-y) &= (x \cdot y) \cdot (-y) \\ &= x \cdot (y \cdot (-y)) \\ &= x \cdot 0 \\ &= 0. \end{aligned}$$

□

Satz 2.36. *Sei $(B, +, \cdot, -, 0, 1)$ eine Boolesche Algebra. Definiere eine 2-stellige Relation $\leqslant$ auf B durch*

$$x \leqslant y \text{ gdw. } x \cdot (-y) = 0 \,(\text{gdw. } x \cdot y = y).$$

Dann ist die Struktur $(B, \leqslant)$ eine partielle Ordnung.

Beweis. *Transitivität*: Betrachte $x, y, z \in B$ mit $x \leqslant y$ und $y \leqslant z$. Dann ist $x \cdot (-y) = 0$ und $y \cdot (-z) = 0$. Hieraus folgt:

$$\begin{aligned} x \cdot (-z) &= (x \cdot 1) \cdot (-z) \\ &= (x \cdot (y + (-y))) \cdot (-z) \\ &= ((x \cdot y) + (x \cdot (-y))) \cdot (-z) \\ &= ((x \cdot y) + 0) \cdot (-z) \\ &= (x \cdot y) \cdot (-z) \\ &= x \cdot (y \cdot (-z)) \\ &= x \cdot 0 \\ &= 0 \end{aligned}$$

Also ist $x \leqslant z$.
Reflexivität: Betrachte $x \in B$. Dann ist $x \cdot (-x) = 0$ und daher $x \leqslant x$.
Antisymmetrie: Betrachte $x, y \in B$ mit $x \leqslant y$ und $y \leqslant x$. Dann ist $x \cdot (-y) = 0$ und $y \cdot (-x) = 0$. Hieraus folgt:

$$\begin{aligned} x &= x \cdot 1 \\ &= x \cdot (y + (-y)) \\ &= (x \cdot y) + (x \cdot (-y)) \\ &= x \cdot y \\ &= (x \cdot y) + 0 \\ &= (x \cdot y) + ((-x) \cdot y) \\ &= (x + (-x)) \cdot y \\ &= 1 \cdot y \\ &= y. \end{aligned}$$

□

Satz 2.37. *Sei $(B, +, \cdot, -, 0, 1)$ eine endliche Boolesche Algebra, d.h. die Trägermenge B ist endlich. Dann ist B isomorph zu einer Potenzmengenalgebra, d.h. es gibt eine Menge A und eine bijektive Abbildung $f\colon B \leftrightarrow \mathrm{Pot}(A)$, die mit den Algebraoperationen verträglich ist:*

a) $f(0) = \emptyset$, $f(1) = A$;

b) $\forall x, y \in B\colon f(x + y) = f(x) \cup f(y)$;

c) $\forall x, y \in B\colon f(x \cdot y) = f(x) \cap f(y)$;

d) $\forall x \in B\colon f(-x) = A \setminus f(x)$.

Beweis. Ein Element $a \in B$ ist ein **Atom** in B, wenn

$$a \neq 0 \wedge \forall x \in B\colon (x \leqslant a \rightarrow (x = 0 \vee x = a)),$$

wobei $\leqslant$ die partielle Ordnung aus dem vorangehenden Satz ist.

(1) Sei $x \in B$ und $x \neq 0$. Dann gibt es ein Atom a in B mit $a \leqslant x$.
Beweis: Angenommen nicht. Definiere dann eine Folge $(x_n | n \in \mathbb{N})$ durch Rekursion: $x_0 = x$. Sei x_n definiert, so dass $x_n \leqslant x$, $x_n \neq 0$. Nach der Widerspruchsannahme ist x_n kein Atom. Wähle dann $x_{n+1} \leqslant x_n$, so dass $x_{n+1} \neq x_n$ und $x_{n+1} \neq 0$. Dann ist $(x_n \,|\, n \in \mathbb{N})$ eine unendliche absteigende Folge in B:

$$x_0 > x_1 > x_2 > \ldots .$$

Das ist ein Widerspruch, da B endlich ist. *qed*

Setze $A = \{a \in B \,|\, B \text{ ist ein Atom in } B\}$. Definiere $f\colon B \to \mathrm{Pot}(A)$ durch

$$f(x) = \{a \in A \,|\, a \leqslant x\}.$$

Wir zeigen die behaupteten Eigenschaften für f.
(2) Seien $a, b \in B$ Atome in B mit $a \cdot b \neq 0$. Dann ist $a = b$.
Beweis: Sei $x = a \cdot b \neq 0$.

$$x \cdot a = (a \cdot b) \cdot a = (b \cdot a) \cdot a = b \cdot (a \cdot a) = b \cdot a = a \cdot b = x$$

und daher ist $x \leqslant a$. Da a ein Atom ist, ist $x = a$. Ebenso ist $x = b$ und $a = b$.
(3) f ist injektiv.
Beweis: Betrachte x, $y \in B$, $x \neq y$. Dann ist $x \not\leqslant y$ oder $y \not\leqslant x$. Ohne Einschränkung der Allgemeinheit sei $x \not\leqslant y$. Nach der Definition von $\leqslant$ ist dann $x \cdot (-y) \neq 0$. Nach (1) wähle ein Atom $a \in A$ mit $a \leqslant x \cdot (-y)$.

$$\begin{aligned} 0 &= a \cdot (-(x \cdot (-y))) \\ &= a \cdot ((-x) + (-(-y))) \\ &= a \cdot ((-x) + y) \\ &= (a \cdot (-x)) + (a \cdot y). \end{aligned}$$

Daraus folgt: $a \cdot (-x) = 0$ und $a \leqslant x$. Weiter ist $a \cdot y = 0$ und

$$a \cdot (-y) = 0 + (a \cdot (-y)) = (a \cdot y) + (a \cdot (-y)) = a \cdot (y + (-y)) = a \cdot 1 = a \neq 0.$$

Daher ist $a \not\leqslant y$. Nach Definition von f ist $a \in f(x)$ und $a \notin f(y)$. Damit ist $f(x) \neq f(y)$. *qed*
(4) f ist surjektiv.
Beweis: Betrachte $X \in \mathrm{Pot}(A)$. Sei $X = \{a_0, \ldots a_{n-1}\}$. Definiere

$$x = a_0 + a_1 + \ldots + a_{n-1} \in B.$$

Wir zeigen, dass $f(x) = X$. Diese Mengengleichheit weisen wir durch zwei Inklusionen nach.
Betrachte $a \in f(x) = \{a \in A \,|\, a \leqslant x\}$. Dann ist

$$a \leqslant x = a_0 + a_1 + \ldots + a_{n-1}$$

und

$$a = a \cdot x = a \cdot (a_0 + a_1 + \ldots + a_{n-1}) = (a \cdot a_0) + \ldots + (a \cdot a_{n-1}).$$

Da $a \neq 0$, wähle ein $i < n$ mit $a \cdot a_i \neq 0$. Nach (2) ist dann $a = a_i$ und $a \in X$. Andererseits sei $a = a_i \in X$. Nach (2) ist dann

$$a \cdot x = a \cdot (a_0 + a_1 + \ldots + a_{n-1}) = (a \cdot a_0) + \ldots + (a \cdot a_{n-1}) = a$$

und $a \leqslant x$. Damit ist $a \in f(x)$. □

Aufgabe 2.1. Beweisen Sie die Teile b)-d) des Satzes.

Kapitel 3
Strukturen

3.1 Signaturen

Wir haben eine Fülle von *Strukturen* kennengelernt: die *Zahlsysteme* $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$ und $\mathbb{C}$; *Körper* und *Vektorräume*; *Ordnungsstrukturen*, *Verbände* und *Boolesche Algebren*. Wir wollen den allgemeinen Strukturbegriff, den wir im ersten Semester eingeführt hatten, weiterentwickeln.

Gemeinsames Merkmal von Strukturen ist das Vorhandensein von *Trägermengen* auf denen *Relationen* oder *Funktionen* wirken. Ein $\mathbb{K}$-Vektorraum V setzt sich aus dem Körper $\mathbb{K}$ mit seiner Trägermenge und den zugehörigen Körperoperationen sowie einer Menge V von Vektoren zusammen. Die Vektoroperationen haben Skalare aus $\mathbb{K}$ oder Vektoren als Argumente:

$$V\colon \begin{cases} \mathbb{K}\colon \begin{cases} K \\ +_{\mathbb{K}}, \cdot_{\mathbb{K}} \ldots \end{cases} \\ V\colon \begin{cases} V \\ +_V, \ldots \end{cases} \end{cases}$$

Die Relationen und Funktionen einer Struktur können sich wie die Skalarmultiplikation auf verschiedene Trägermengen beziehen und können verschiedene *Stellenzahlen* haben. Zur Organisation des Systems von Trägermengen und Relationen und Funktionen führen wir allgemein *Signaturen* ein. Eine Signatur liefert *Symbole* zur Referenzierung verschiedener Strukturkomponenten und fixiert die Stellenzahlen von Relationen und Funktionen.

Definition 3.1. *Ein 5-Tupel* $\sigma = (S, F, R, K, \mathrm{fct})$ *ist eine* ***Signatur****, wenn*

a) S, F, R, K *sind paarweise disjunkte Mengen von* ***Sorten****,* ***Funktionssymbolen****,* ***Relationssymbolen*** *bzw.* ***Konstantensymbolen****;*

b) fct *ist eine auf* $F \cup R \cup K$ *definierte Funktion, die als* ***Funktionalität*** *bezeichnet wird;*

c) für alle $f \in F$ *gibt es ein* $n \in \mathbb{N}$ *mit* $\mathrm{fct}(f) \in S^{n+1}$*;*

d) für alle $r \in R$ *gibt es ein* $n \in \mathbb{N}$ *mit* $\mathrm{fct}(r) \in S^n$*;*

e) für alle $k \in K$ *ist* $\mathrm{fct}(k) \in S$*.*

Bemerkung 3.2. Die Sorten entsprechen den verschiedenen Trägermengen von Strukturen. Die Funktion fct legt die *Typen* der Symbole fest. Wenn $f \in F$ ein Funktionssymbol ist und $\mathrm{fct}(f) = (s_0, \ldots, s_{n-1}, s_n)$, so bedeutet das, dass f n-stellig ist, seine Argumente aus den mit den Sorten s_0, ..., s_{n-1} bezeichneten Trägermengen bezieht und einen Wert in der mit s_n bezeichneten Trägermenge liefert.

$\mathbb{K}$-Vektorräume $V = (V, +, \cdot, 0)$ kann man folgendermaßen durch eine Signatur σ_{VR} erfassen:

Sortenmenge

$$S = \{\text{Vektor}, \text{Skalar}\};$$

als Operationen liegen Körperoperationen $+_{\mathbb{K}}$ und $\cdot_{\mathbb{K}}$, die Vektoraddition $+_V$ und die Skalarmultiplikation $\cdot_V$ vor:

$$F = \{ +_{\mathbb{K}}, \cdot_{\mathbb{K}}, +_V, \cdot_V \};$$

Relationen liegen nicht vor:

$$R = \emptyset;$$

es gibt die Körperkonstanten $0_{\mathbb{K}}$ und $1_{\mathbb{K}}$ und den Nullvektor 0_V:

$$K = \{0_{\mathbb{K}}, 1_{\mathbb{K}}, 0_V\}$$

Die Funktionalität der verschiedenen Symbole in $F \cup R \cup K$ ist:

$$\begin{aligned}
\mathrm{fct}(+_{\mathbb{K}}) &= (\text{Skalar}, \text{Skalar}, \text{Skalar}) \\
\mathrm{fct}(\cdot_{\mathbb{K}}) &= (\text{Skalar}, \text{Skalar}, \text{Skalar}) \\
\mathrm{fct}(+_V) &= (\text{Vektor}, \text{Vektor}, \text{Vektor}) \\
\mathrm{fct}(\cdot_V) &= (\text{Skalar}, \text{Vektor}, \text{Vektor}) \\
\mathrm{fct}(0_{\mathbb{K}}) &= \text{Skalar} \\
\mathrm{fct}(1_{\mathbb{K}}) &= \text{Skalar} \\
\mathrm{fct}(0_V) &= \text{Vektor}.
\end{aligned}$$

Das bedeutet zum Beispiel, dass $+_{\mathbb{K}}$ eine Operation beschreibt, die von $\mathbb{K}^2$ nach $\mathbb{K}$ geht, dass $\cdot_V$ von $\mathbb{K} \times V$ nach V geht, und dass 0_V ein Element von V ist. Damit ist

$$\sigma_{\mathrm{VR}} = (\{\text{Vektor}, \text{Skalar}\}, \{ +_{\mathbb{K}}, \cdot_{\mathbb{K}}, +_V, \cdot_V \}, \emptyset, \{0_{\mathbb{K}}, 1_{\mathbb{K}}, 0_V\}, \mathrm{fct}).$$

Die Funktionalitäten von Funktionssymbolen werden auch folgendermaßen geschrieben:

$$\begin{aligned}
\mathrm{fct}(+_V) &= (\text{Vektor}, \text{Vektor})\,\text{Vektor} \\
\mathrm{fct}(\cdot_V) &= (\text{Skalar}, \text{Vektor})\,\text{Vektor}.
\end{aligned}$$

Derartige Konstrukte finden sich auch in Programmiersprachen. Eine Signatur entspricht der *Klassendeklaration* in einer Programmiersprache wie C++. Wenn man Vektorräume als *Klasse* einführen möchte, würde man unter der Annahme, dass die Datentypen `vektor` und `skalar` vorliegen, etwa schreiben:

```
class Vektorraum
{
public:
vektor Vektoraddition(vektor,vektor);
vektor Skalarmultiplikation(skalar,vektor);
vektor Nullvektor();
```

```
};
```

3.2 Strukturen

Eine Signatur kann durch *Strukturen* interpretiert werden. In einer Struktur werden den Symbolen entsprechende Strukturkomponenten zugeordnet.

Definition 3.3. *Sei* $\sigma = (S,\ F,\ R,\ K,\ \mathrm{fct})$ *eine Signatur. Eine* ***Struktur*** *mit Signatur oder eine* σ-***Struktur*** *ist ein Tupel*

$$\mathfrak{A} = ((A_s)_{s\in S}, (f^A)_{f\in F}, (r^A)_{r\in R}, (k^A)_{k\in K})$$

mit den Eigenschaften:

a) *für* $s \in S$ *ist* $A_s \neq \emptyset$*; jedes* A_s *ist eine* ***Trägermenge*** *der Struktur;*

b) *für* $f \in F$ *mit* $\mathrm{fct}(f) = (s_0, ..., s_{n-1}, s_n)$ *ist* f^A *eine* ***Funktion****:*

$$f^A\colon A_{s_0} \times ... \times A_{s_{n-1}} \to A_{s_n}\,;$$

c) *für* $r \in R$ *mit* $\mathrm{fct}(r) = (s_0, ..., s_{n-1})$ *ist* r^A *eine* ***Relation****:*

$$r^A \subseteq A_{s_0} \times ... \times A_{s_{n-1}}\,;$$

d) *für* $k \in K$ *mit* $\mathrm{fct}(k) = s$ *ist* k^A *eine* ***Konstante****:*

$$k^A \in A_s\,.$$

Die Struktur $\mathfrak{A}$ kann als Zuordnung von Symbolen zu (mathematischen) Objekten geeigneten Typs aufgefasst werden:

$$s \mapsto A_s\,,\ f \mapsto f^A\,,\ r \mapsto r^A\,,\ k \mapsto k^A.$$

Im Fall der oben diskutierten Vektorräume kann ein gewöhnlicher Vektorraum als Struktur mit der Signatur

$$\sigma_{\mathrm{VR}} = (\{\mathrm{Vektor}, \mathrm{Skalar}\}, \{+_{\mathrm{Sk}}\,, \cdot_{\mathrm{Sk}}\,, +_{\mathrm{Vk}}\,, \cdot_{\mathrm{Vk}}\}, \emptyset, \{0_{\mathrm{Sk}}, 1_{\mathrm{Sk}}, 0_{\mathrm{Vk}}\}, \mathrm{fct})$$

aufgefasst werden; der 2-dimensionale $\mathbb{R}$-Vektorraum $\mathbb{R}^2$ wäre in dieser Formatierung:

$$\mathbb{R}^2 = (\{\mathbb{R}^2, \mathbb{R}\}, \{+_{\mathbb{R}}\,, \cdot_{\mathbb{R}}\,, +_{\mathbb{R}^2}\,, \cdot_{\mathbb{R}^2}\}, \emptyset, \{0, 1, \begin{pmatrix} 0 \\ 0 \end{pmatrix}\}.$$

Formulierungen dieser Art enthalten in der Regel viele redundante Elemente, daher werden sie etwa verkürzt zu dem bekannten

$$\mathbb{R}^2 = (\mathbb{R}^2, +_{\mathbb{R}^2}\,, \cdot_{\mathbb{R}^2}\,, \begin{pmatrix} 0 \\ 0 \end{pmatrix}\}.$$

Da der Skalarkörper $\mathbb{R}$ fixiert ist, werden seine Komponenten unterdrückt; auch die leere Menge der Relationen braucht üblicherweise nicht angegeben zu werden. Weiterhin wird eine Struktur häufig durch (eine) ihre(r) Trägermengen bezeichnet.

Dennoch sollte man sich bewusst sein, dass die üblichen Abkürzungsschreibweisen bei Bedarf zu vollständigen, eindeutigen Formulierungen ausgedehnt werden können.

Aufgabe 3.1. Definieren Sie eine Signatur σ_{arith} der *Arithmetik*, die für verschiedene Zahlsysteme wie $\mathbb{N}$, $\mathbb{Q}$, $\mathbb{R}$ mit Addition und Multiplikation adäquat ist. Stellen Sie die Zahlbereiche als Strukturen mit dieser Signatur dar.

Aufgabe 3.2. Definieren Sie Signaturen für *Booleschen Algebren* entsprechend der zwei alternativen Definitionen von Boolescher Algebra und stellen Sie die Potenzmengenalgebra $\mathcal{P}(A)$ als Strukturen zu diesen Signaturen dar.

3.3 Substrukturen

Zu einer gegebenen Signatur σ gibt es eine Fülle von σ-Strukturen. In der Linearen Algebra hatten wir etwa eine große Vielfalt von Vektorräumen kennengelernt. Eine Aufgabe der Theorie ist es, durch Klassifizierungen einen Überblick über die Klasse aller Möglichkeiten zu erlangen. Bei Vektorräumen waren hierzu Begriffe wie *Unterraum* und *lineare Abbildung* eingeführt worden. Wir wollen derartige Definitionen ganz allgemein studieren:

Definition 3.4. *Sei* $\sigma = (S, F, R, K, \text{fct})$ *eine Signatur und seien*

$$\mathfrak{A} = ((A_s)_{s\in S}, (f^A)_{f\in F}, (r^A)_{r\in R}, (k^A)_{k\in K})$$

und

$$\mathfrak{B} = ((B_s)_{s\in S}, (f^B)_{f\in F}, (r^B)_{r\in R}, (k^B)_{k\in K})$$

*σ-Strukturen. Dann ist $\mathfrak{A}$ **Substruktur** oder **Unterstruktur** von $\mathfrak{B}$, wenn:*

a) für $s \in S$ ist $A_s \subseteq B_s$;

b) für $f \in F$ mit $\text{fct}(f) = (s_0, ..., s_{n-1}, s_n)$ und $a_0 \in A_{s_0}, ..., a_{n-1} \in A_{s_{n-1}}$ ist

$$f^A(a_0, ..., a_{n-1}) = f^B(a_0, ..., a_{n-1});$$

c) für $r \in R$ mit $\text{fct}(r) = (s_0, ..., s_{n-1})$ und $a_0 \in A_{s_0}, ..., a_{n-1} \in A_{s_{n-1}}$ ist

$$r^A(a_0, ..., a_{n-1}) \text{ gdw. } r^B(a_0, ..., a_{n-1});$$

d) für $k \in K$ mit $\text{fct}(k) = s$ ist

$$k^A = k^B.$$

Man schreibt dann auch $\mathfrak{A} \subseteq \mathfrak{B}$.

Eine Substruktur wird durch *Einschränkung* der Trägermengen gegeben; die übrigen Komponenten der Struktur werden entsprechend eingeschränkt. Ein Untervektorraum U eines $\mathbb{K}$-Vektorraums V ist eine Substruktur im Sinne dieser Definition:

$$U = (\{U, \mathbb{K}\}, \{+_{\mathbb{K}}, \cdot_{\mathbb{K}}, +_U, \cdot_U\}, \emptyset, \{0_{\mathbb{K}}, 1_{\mathbb{K}}, 0_U\}) \quad \subseteq$$
$$V = (\{V, \mathbb{K}\}, \{+_{\mathbb{K}}, \cdot_{\mathbb{K}}, +_V, \cdot_V\}, \emptyset, \{0_{\mathbb{K}}, 1_{\mathbb{K}}, 0_V\}).$$

3.4 Homomorphismen

Im Mittelpunkt der Linearen Algebra steht das Studium der *linearen Abbildungen*. Dieses sind Abbildungen zwischen den Trägermengen der Vektoren, die mit beiden Strukturen verträglich sind. Wir formulieren allgemein:

Definition 3.5. *Sei* $\sigma = (S, F, R, K, \mathrm{fct})$ *eine Signatur und seien*

$$\mathfrak{A} = ((A_s)_{s\in S}, (f^A)_{f\in F}, (r^A)_{r\in R}, (k^A)_{k\in K})$$

und

$$\mathfrak{B} = ((B_s)_{s\in S}, (f^B)_{f\in F}, (r^B)_{r\in R}, (k^B)_{k\in K})$$

σ-Strukturen. Sei $s \in S$, so dass für alle $s' \in S \setminus \{s\}$ gilt: $A_{s'} = B_{s'}$. Für eine Abbildung $h\colon A_s \to B_s$ und $t \in S$ definiere

$$h_t\colon A_t \to B_t\,,\ h_t = \begin{cases} h, & \text{für } t = s \\ \mathrm{Id}_{A_t}, & \text{für } t \neq s \end{cases}$$

*Dann ist $h\colon A_s \to B_s$ ein s-**Homomorphismus** von $\mathfrak{A}$ nach $\mathfrak{B}$, wenn:*

a) für $f \in F$ mit $\mathrm{fct}(f) = (s_0, ..., s_{n-1}, s_n)$ und $a_0 \in A_{s_0}, ..., a_{n-1} \in A_{s_{n-1}}$ ist

$$f^B(h_{s_0}(a_0), ..., h_{s_{n-1}}(a_{n-1})) = h_{s_n}(f^A(a_0, ..., a_{n-1}));$$

b) für $r \in R$ mit $\mathrm{fct}(r) = (s_0, ..., s_{n-1})$ und $a_0 \in A_{s_0}, ..., a_{n-1} \in A_{s_{n-1}}$ ist

$$r^B(h_{s_0}(a_0), ..., h_{s_{n-1}}(a_{n-1})) \text{ gdw. } r^A(a_0, ..., a_{n-1});$$

c) für $k \in K$ mit $\mathrm{fct}(k) = s_0$ ist

$$k^B = h_{s_0}(k^A).$$

Man schreibt dann auch $h\colon \mathfrak{A} \to_s \mathfrak{B}$ oder einfacher $h\colon \mathfrak{A} \to \mathfrak{B}$.

Beispiel 3.6. Sei

$$\sigma_{\mathrm{VR}} = (\{\mathrm{Vektor}, \mathrm{Skalar}\}, \{+_{\mathrm{Sk}}, \cdot_{\mathrm{Sk}}, +_{\mathrm{Vk}}, \cdot_{\mathrm{Vk}}\}, \emptyset, \{0_{\mathrm{Sk}}, 1_{\mathrm{Sk}}, 0_{\mathrm{Vk}}\}, \mathrm{fct})$$

die Signatur der Vektorräume und seien

$$U = (\{U, \mathbb{K}\}, \{+_{\mathbb{K}}, \cdot_{\mathbb{K}}, +_U, \cdot_U\}, \emptyset, \{0_{\mathbb{K}}, 1_{\mathbb{K}}, 0_U\})$$

und

$$V = (\{V, \mathbb{K}\}, \{+_{\mathbb{K}}, \cdot_{\mathbb{K}}, +_V, \cdot_V\}, \emptyset, \{0_{\mathbb{K}}, 1_{\mathbb{K}}, 0_V\})$$

$\mathbb{K}$-Vektorräume. Eine Abbildung $h\colon U \to V$ ist genau dann ein Vektor-Homomorphismus, wenn h eine lineare Abbildung von $\mathbb{K}$-Vektorräumen ist. Es gilt $h_{\mathrm{Vektor}} = h$ und $h_{\mathrm{Skalar}} = \mathrm{Id}_{\mathbb{K}}$. Die Homomorphismuseigenschaft für das Skalarprodukt eines Skalars $\lambda \in \mathbb{K}$ und eines Vektors $u \in U$ bedeutet:

$$\cdot_V(h_{\mathrm{Skalar}}(\lambda), h_{\mathrm{Vektor}}(u)) \;=\; h_{\mathrm{Vektor}}(\cdot_U(\lambda, u))$$

Dies kann äquivalent zur multiplikativen Linearität von h umgeformt werden:

$$\begin{aligned} \cdot_V(\lambda, h(u)) &= h(\cdot_U(\lambda, u)) \\ \lambda \cdot_V h(u) &= h(\lambda \cdot_U u) \\ \lambda \cdot h(u) &= h(\lambda \cdot u). \end{aligned}$$

Wir führen Begriffe zur Beschreibung von Homomorphismen ein:

Definition 3.7. *Sei $h\colon \mathfrak{A} \to_s \mathfrak{B}$ ein s-Homomorphismus zwischen σ-Strukturen $\mathfrak{A}$ und $\mathfrak{B}$. Dann ist h eine Abbildung $h\colon A_s \to B_s$ zwischen den Trägermengen A_s und B_s. Wir definieren:*

*a) h ist ein **Monomorphismus**, wenn h injektiv ist;*

*b) h ist ein **Epimorphismus**, wenn h surjektiv ist;*

*c) h ist ein **Isomorphismus**, wenn h bijektiv ist;*

*d) h ist ein **Endomorphismus**, wenn $A_s = B_s$;*

*e) h ist ein **Automorphismus**, wenn h bijektiv und $A_s = B_s$ ist.*

Kapitel 4
Sprachen

Formale Sprache dienen dazu, Aussagen über Strukturen einer vorgegebenen Signatur zu machen. Die Aussage $\forall x \forall y$: $x + y = y + x$ besagt zum Beispiel, dass die Addition (in Strukturen geeigneter Signatur) kommutativ ist. Eine Signatur stellt bereits Mengen von Symbolen zur Verfügung, die bei der Bildung von Aussagen der Sprache verwendet werden können. Diese Symbole werden durch weitere Symbole in logische Zusammenhänge gestellt. Die formale Sprache wird durch ihre Signatur bestimmt. Die formalen Aspekte der Sprache bezeichnet man als *Syntax*. Die Interpretation einer Sprache in Strukturen ist Anliegen der *Semantik*.

4.1 Syntax

Definition 4.1. *Sei* $\sigma = (S,\ F,\ R,\ K,\ \mathrm{fct})$ *eine Signatur. Die zu* σ *gehörige Sprache* L^σ *besteht aus mehreren Komponenten:*

a) ***Variable****: für jede Sorte* $s \in S$ *gibt es Variablen* $v_0^s, v_1^s, \ldots$*;*

b) ***Symbole****: die Symbolmenge* A^σ *der Sprache* L^σ *ist*

$$A^\sigma = F \cup R \cup K \cup \{v_n^s \mid s \in S, n \in \mathbb{N}\} \cup \{(,),'',',=,\neg,\wedge,\vee,\rightarrow,\leftrightarrow,\forall,\exists\}$$

c) ***Terme****: Die Menge* T^σ *der* σ*-Terme wird rekursiv definiert. Jedem Term* t *wird außerdem ein* ***Typ*** $\tau(t) \in S$ *zugeordnet:*

i. alle Variablen v_n^s *sind Terme vom Typ* $\tau(v_n^s) = s$*;*

ii. alle Konstantensymbole $k \in K$ *sind Terme vom Typ* $\tau(k) = \mathrm{fct}(k)$*;*

iii. für alle n-stelligen Funktionssymbole $f \in F$ *mit der Funktionalität*

$$\mathrm{fct}(f) = (s_0, \ldots, s_{n-1}, s_n)$$

und Terme $t_0, \ldots, t_{n-1}$ *mit* $\tau(t_0) = s_0, \ldots, \tau(t_{n-1}) = s_{n-1}$ *ist*

$$f(t_0, \ldots, t_{n-1})$$

ein Term mit $\tau(f(t_0, \ldots, t_{n-1})) = s_n$*.*

d) ***Relationale Aussagen****: für eine Sorte* $s \in S$ *und Terme* $t_0,\ t_1 \in T^\sigma$ *mit* $\tau(t_0) = \tau(t_1) = s$ *ist*

$$t_0 = t_1$$

eine relationale Aussage; und für alle n-stelligen Relationssymbole $r \in R$ mit der Funktionalität

$$\mathrm{fct}(r) = (s_0, ..., s_{n-1})$$

und Terme $t_0, ..., t_{n-1}$ mit $\tau(t_0) = s_0, ..., \tau(t_{n-1}) = s_{n-1}$ ist

$$r(t_0, ..., t_{n-1})$$

eine relationale Aussage.

e) ***Aussagen**: Die Menge* Aus^σ *der Sprache L^σ wird rekursiv definiert:*

i. jede relationale Aussage ist eine Aussage;

ii. wenn φ eine Aussage ist, so ist auch $\neg\varphi$ („nicht φ") eine Aussage;

iii. wenn φ und ψ Aussagen sind, so sind auch

$$\begin{array}{ll} (\varphi \wedge \psi) & \text{„}\varphi \text{ und } \psi\text{"} \\ (\varphi \vee \psi) & \text{„}\varphi \text{ oder } \psi\text{"} \\ (\varphi \to \psi) & \text{„}\varphi \text{ impliziert } \psi\text{"} \\ (\varphi \leftrightarrow \psi) & \text{„}\varphi \text{ ist äquivalent zu } \psi\text{"} \end{array}$$

Aussagen;

iv. wenn φ eine Aussage ist und v_n^s eine Variable, so sind auch

$$\begin{array}{ll} \forall v_n^s \varphi & \text{„für alle } v_n^s \text{ gilt } \varphi\text{"} \\ \exists v_n^s \varphi & \text{„es gibt ein } v_n^s \text{ mit } \varphi\text{"} \end{array}$$

Aussagen.

Beispiel 4.2. *Die Sprache der Vektorräume.* Wir wenden die Definition an und führen verschiedene abkürzende Schreibweisen ein. Sei

$$\begin{aligned} \sigma_{\mathrm{VR}} &= (\{\mathrm{Vektor}, \mathrm{Skalar}\}, \{ +_{\mathrm{Sk}}, \cdot_{\mathrm{Sk}}, +_{\mathrm{Vk}}, \cdot_{\mathrm{Vk}} \}, \emptyset, \{0_{\mathrm{Sk}}, 1_{\mathrm{Sk}}, 0_{\mathrm{Vk}}\}, \mathrm{fct}) \\ &= (\mathrm{Vektor}, \mathrm{Skalar}; +_{\mathrm{Sk}}, \cdot_{\mathrm{Sk}}, +_{\mathrm{Vk}}, \cdot_{\mathrm{Vk}}, 0_{\mathrm{Sk}}, 1_{\mathrm{Sk}}, 0_{\mathrm{Vk}}, \mathrm{fct}) \end{aligned}$$

die Signatur der Vektorräume. Dann hat man:

– *Variablen*: v_n^{Vektor} und v_n^{Skalar}, die üblicherweise durch besondere Buchstaben x, y, z, ... für Vektor-Variablen und λ, μ, ν, ... für Skalar-Variablen bezeichnet werden;

– *Terme*: $+_{\mathrm{Sk}}(\lambda, \mu)$, $\cdot_{\mathrm{Sk}}(\lambda, \mu)$, $+_{\mathrm{Vk}}(x, y)$, $\cdot_{\mathrm{Vk}}(\lambda, x)$ und komplexere Terme der Gestalt

$$+_{\mathrm{Vk}}(\cdot_{\mathrm{Vk}}(\lambda, x), \cdot_{\mathrm{Vk}}(\mu, y)).$$

Die Terme sind in üblichen arithmetischen Infix-Schreibweisen mit Klammersetzung besser zu verstehen: $\lambda +_{\mathrm{Sk}} \mu, \lambda \cdot_{\mathrm{Sk}} \mu, ...$ und

$$(\lambda \cdot_{\mathrm{Vk}} x) +_{\mathrm{Vk}} (\mu \cdot_{\mathrm{Vk}} y).$$

Da die Variablen den Typ der auf sie anwendbaren Funktionen bestimmen, lässt sich ableiten, ob an bestimmten Stellen des Terms $+_{\mathrm{Sk}}$ oder $\cdot_{\mathrm{Vk}}$ stehen muss. Daher kann man die Indizes Sk oder Vk in der Regel fortlassen:

$$(\lambda \cdot x) + (\mu \cdot y).$$

Mit Konventionen wie „Punktrechnung geht vor Strichrechnung" kann man weiterhin Klammern um die Skalar-Multiplikationen fortlassen. Außerdem wird der Mal-Punkt $\cdot$ gern eliminiert:

$$\lambda x + \mu y.$$

– *Relationale Aussagen*: Als relationales Symbol liegt nur das Gleichheitszeichen $=$ vor, daher kommen nur Aussagen der Art

$$+_{\mathrm{Vk}}(\cdot_{\mathrm{Vk}}(\lambda, x), \cdot_{\mathrm{Vk}}(\mu, y)) = 0_{\mathrm{Vk}}$$

in Frage, die selbstverständlich vertrauter als

$$\lambda x + \mu y = 0$$

geschrieben werden. Man beachte, dass die Konstante 0 auf der rechten Seite vom Typ Vektor sein muss, da die linke Seite der Gleichung diesen Typ hat. Daher können wir einfach 0 statt 0_{Vk} schreiben.

– *Aussagen*: Die Vektorraum-Axiome sind eine endliche Menge von Aussagen, die für alle Vektorräume gefordert werden. In der formalen Sprache lautet eines der Assoziativgesetze:

$$\forall v_0^{\mathrm{Skalar}} \forall v_1^{\mathrm{Skalar}} \forall v_0^{\mathrm{Vektor}} \quad \cdot_{\mathrm{Vk}}(v_0^{\mathrm{Skalar}}, \cdot_{\mathrm{Vk}}(v_1^{\mathrm{Skalar}}, v_0^{\mathrm{Vektor}})) = \cdot_{\mathrm{Vk}}(\cdot_{\mathrm{Sk}}(v_0^{\mathrm{Skalar}}, v_1^{\mathrm{Skalar}}), v_0^{\mathrm{Vektor}}).$$

Unter der Benutzung obiger Konventionen wird hieraus:

$$\forall\lambda\forall\mu\forall x\, \lambda \cdot (\mu \cdot x) = (\lambda \cdot \mu) \cdot x.$$

Assoziativgesetze erlauben anschließend das Fortlassen weiterer Klammern, da sie gerade die Unabhängigkeit von Werten von der Klammersetzung zum Inhalt haben. Manchmal deutet man die Typen der Variablen durch zusätzliche mengentheoretische Formulierungen an. Für einen $\mathbb{K}$-Vektorraum V schreibt man das Assoziativgesetz auch als:

$$\forall\lambda \in \mathbb{K}\, \forall\mu \in \mathbb{K}\, \forall x \in V\, \lambda \cdot (\mu \cdot x) = (\lambda \cdot \mu) \cdot x.$$

Bemerkung 4.3. In der Informatik besteht die ständige Notwendigkeit, adäquate Sprachen zu definieren. In der verbreiteten Backus-Naur-Form (BNF) lässt sich die obige Sprache folgendermaßen beschreiben:

Variablen:	V	$::=$	v_n^s	$n \in \mathbb{N}, s \in S$
Terme:	T	$::=$	$V \mid f(V_0, ..., V_{n-1})$	$f \in F$
Relationale Aussagen:	RA	$::=$	$T_0 = T_1 \mid r(T_0, ..., T_{n-1})$	$r \in R$
Aussagen:	A	$::=$	$\mathrm{RA} \mid \neg A \mid (A_0 \wedge A_1) \mid (A_0 \vee A_1) \mid \mid (A_0 \to A_1) \mid (A_0 \leftrightarrow A_1)$	

Allerdings werden hier Details wie die Beachtung von Typen bei der Bildung von Termen und Aussagen unterdrückt. Durch die BNF-Definition der formalen Sprache L^σ wird die Ähnlichkeit zu Programmiersprachen betont.

4.2 Semantik

Mit der Sprache L^σ kann man Eigenschaften von σ-Strukturen formulieren. Durch Interpretation der Elemente der Sprache in einer σ-Struktur kann man überprüfen, ob die Struktur die Eigenschaft erfüllt. Die Definition erfolgt rekursiv über den Aufbau der formalen Sprache. Neben den Symbolen aus σ benutzt die formale Sprache auch Variablen für Elemente der Trägermengen. Die Variablen werden interpretiert, indem sie mit Werten aus den Trägermengen *belegt* werden.

Definition 4.4. *Sei* $\sigma = (S, F, R, K, \mathrm{fct})$ *eine Signatur mit zugehöriger Sprache* L^σ. *Sei*

$$\mathfrak{A} = ((A_s)_{s\in S}, (f^A)_{f\in F}, (r^A)_{r\in R}, (k^A)_{k\in K})$$

eine σ*-Struktur. Die Interpretation von* L^σ *in* $\mathfrak{A}$ *wird schrittweise definiert:*

- *Eine* ***Belegung*** *in* $\mathfrak{A}$ *ist eine Funktion*

$$\beta\colon \{v_n^s \mid n\in\mathbb{N}, s\in S\} \to \bigcup_{s\in S} A_s\ ,$$

so dass für alle $n\in\mathbb{N}$ *und* $s\in S$ *gilt* $\beta(v_n^s)\in A_s$. *Die Belegung interpretiert also Variablen entsprechend ihrem Typ.*
Es ist manchmal wichtig, den Wert einer Belegung β *an einer Variablen* $v_{n'}^{s'}$ *zu einem gegebenen* $a\in A_s$ *zu modifizieren. Definiere dazu eine modifizierte Belegung*

$$\beta\frac{a}{v_{n'}^{s'}}\colon \{v_n^s \mid n\in\mathbb{N}, s\in S\} \to \bigcup_{s\in S} A_s$$

durch

$$\beta\frac{a}{v_{n'}^{s'}}(v_n^s) = \begin{cases} \beta(v_n^s), \text{ falls } v_n^s \neq v_{n'}^{s'} \\ a, \text{ falls } v_n^s = v_{n'}^{s'} \end{cases}$$

- *Ein* σ*-****Modell*** *ist ein geordnetes Paar* $\mathfrak{M} = (\mathfrak{A}, \beta)$ *aus einer* σ*-Struktur* $\mathfrak{A}$ *und einer Belegung* β *in* $\mathfrak{A}$. *Für die weiteren Definitionen sei ein Modell* $\mathfrak{M} = (\mathfrak{A}, \beta)$ *fixiert.*
- *Für eine Term* $t\in T^\sigma$ *der Sprache* L^σ *definiere die* ***Interpretation*** $\mathfrak{M}(t)$ *im Modell* $\mathfrak{M}$ *durch Rekursion über den Aufbau von* t*:*

  i. für eine Variable v_n^s *ist* $\mathfrak{M}(v_n^s) = \beta(v_n^s)$;

  ii. für ein Konstantensymbol $k\in K$ *ist* $\mathfrak{M}(k) = k^A$;

iii. für ein n-stelliges Funktionssymbol $f \in F$ und Terme $t_0, ..., t_{n-1} \in T^\sigma$ ist

$$\mathfrak{M}(f(t_0, ..., t_{n-1})) = f^A(\mathfrak{M}(t_0), ..., \mathfrak{M}(t_{n-1})).$$

– *Für eine Aussage $\varphi \in L^\sigma$ definiere, dass $\mathfrak{M}$ ein **Modell von** φ ist, $\mathfrak{M} \vDash \varphi$, durch Rekursion über den Aufbau von φ:*

i. für Terme $t_0, t_1 \in T^\sigma$ setze

$$\mathfrak{M} \vDash t_0 = t_1 \text{ gdw. } \mathfrak{M}(t_0) = \mathfrak{M}(t_1);$$

ii. für ein n-stelliges Relationssymbol $r \in R$ und Terme $t_0, ..., t_{n-1} \in T^\sigma$ setze

$$\mathfrak{M} \vDash r(t_0, ..., t_{n-1}) \text{ gdw. } r^A(\mathfrak{M}(t_0), ..., \mathfrak{M}(t_{n-1}));$$

iii. $\mathfrak{M} \vDash \neg\varphi$ *gdw.* ***nicht*** $\mathfrak{M} \vDash \varphi$;

iv. $\mathfrak{M} \vDash (\varphi \wedge \psi)$ *gdw.* $\mathfrak{M} \vDash \varphi$ ***und*** $\mathfrak{M} \vDash \psi$;

v. $\mathfrak{M} \vDash (\varphi \vee \psi)$ *gdw.* $\mathfrak{M} \vDash \varphi$ ***oder*** $\mathfrak{M} \vDash \psi$;

vi. $\mathfrak{M} \vDash (\varphi \to \psi)$ *gdw.* $\mathfrak{M} \vDash \varphi$ ***impliziert*** $\mathfrak{M} \vDash \psi$;

vii. $\mathfrak{M} \vDash (\varphi \leftrightarrow \psi)$ *gdw.* $\mathfrak{M} \vDash \varphi$ ***ist äquivalent zu*** $\mathfrak{M} \vDash \psi$;

viii. $\mathfrak{M} \vDash \forall v_n^s \varphi$ *gdw.* ***für alle*** $a \in A_s$ *gilt* $\mathfrak{M}\frac{a}{v_n^s} = (\mathfrak{M}, \beta\frac{a}{v_n^s}) \vDash \varphi$;

ix. $\mathfrak{M} \vDash \exists v_n^s \varphi$ *gdw.* ***es existiert ein*** $a \in A_s$ *mit* $\mathfrak{M}\frac{a}{v_n^s} = (\mathfrak{M}, \beta\frac{a}{v_n^s}) \vDash \varphi$.

*Man sagt auch $\mathfrak{M}$ **erfüllt** φ oder φ **gilt in** $\mathfrak{M}$ für $\mathfrak{M} \vDash \varphi$.*

Bemerkung 4.5. Bei den Definitionen wird benutzt, dass Terme und Aussagen der Sprache *eindeutig lesbar* sind, d.h. für jeden Term gibt es genau eine Weise, wie er in konstituierende Teilterme zerfällt; die Interpretation des Terms wird dann aus den Interpretationen dieser Teilterme zusammengesetzt. Entsprechendes gilt für Aussagen. Formale Sprachen sind so zu konstruieren, dass eindeutige Lesbarkeit gegeben ist. Programme und ihre Konstituenten sollen eindeutig lesbar sein, andernfalls gäbe es Probleme mit der Determiniertheit von Übersetzungen und Programmausführungen.

Beispiel 4.6. Die Definition von Term-Interpretationen und Modell-Beziehung stimmen mit dem bisherigen informellen Gebrauch überein. Dies liegt an den naheliegenden Definitionen für aussagenlogische Verknüpfungen und dem natürlichen Zusammenspiel der verschiedenen Definitionen.

Es sei beispielsweise

$$\sigma_{\mathrm{VR}} = (\mathrm{Vk}, \mathrm{Sk}; +_{\mathrm{Sk}}, \cdot_{\mathrm{Sk}}, +_{\mathrm{Vk}}, \cdot_{\mathrm{Vk}}, 0_{\mathrm{Sk}}, 1_{\mathrm{Sk}}, 0_{\mathrm{Vk}}, \mathrm{fct})$$

die Signatur der Vektorräume und

$$\mathfrak{V} = (V, \mathbb{K}; +_{\mathbb{K}}, \cdot_{\mathbb{K}}, +_V, \cdot_V, 0_{\mathbb{K}}, 1_{\mathbb{K}}, 0_V)$$

ein Vektorraum mit Skalarkörper $\mathbb{K}$. Dann ist

$$\mathfrak{V} \vDash \forall\lambda\forall x\forall y\, \lambda\cdot(x+y) = \lambda\cdot x + \lambda\cdot y$$

gdw. $\mathfrak{V} \vDash \forall v_0^{\mathrm{Sk}}\forall v_0^{\mathrm{Vk}}\forall v_1^{\mathrm{Vk}}\; v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} (v_0^{\mathrm{Vk}} +_{\mathrm{Vk}} v_1^{\mathrm{Vk}}) = v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} v_0^{\mathrm{Vk}} +_{\mathrm{Vk}} v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} v_1^{\mathrm{Vk}}$

gdw. für alle $a \in \mathbb{K}$ gilt:
$\mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}} \vDash \forall v_0^{\mathrm{Vk}}\forall v_1^{\mathrm{Vk}}\, v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} (v_0^{\mathrm{Vk}} +_{\mathrm{Vk}} v_1^{\mathrm{Vk}}) = v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} v_0^{\mathrm{Vk}} +_{\mathrm{Vk}} v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} v_1^{\mathrm{Vk}}$

gdw. für alle $a \in \mathbb{K}$ gilt: für alle $b \in V$ gilt:
$\mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}}\frac{b}{v_0^{\mathrm{Vk}}} \vDash \forall v_1^{\mathrm{Vk}}\, v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} (v_0^{\mathrm{Vk}} +_{\mathrm{Vk}} v_1^{\mathrm{Vk}}) = v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} v_0^{\mathrm{Vk}} +_{\mathrm{Vk}} v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} v_1^{\mathrm{Vk}}$

gdw. für alle $a \in \mathbb{K}$ gilt: für alle $b \in V$ gilt: für alle $c \in V$ gilt:
$\mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}}\frac{b}{v_0^{\mathrm{Vk}}}\frac{c}{v_1^{\mathrm{Vk}}} \vDash v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} (v_0^{\mathrm{Vk}} +_{\mathrm{Vk}} v_1^{\mathrm{Vk}}) = v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} v_0^{\mathrm{Vk}} +_{\mathrm{Vk}} v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} v_1^{\mathrm{Vk}}$

gdw. für alle $a \in \mathbb{K}$ gilt: für alle $b \in V$ gilt: für alle $c \in V$ gilt:
$\mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}}\frac{b}{v_0^{\mathrm{Vk}}}\frac{c}{v_1^{\mathrm{Vk}}}(v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} (v_0^{\mathrm{Vk}} +_{\mathrm{Vk}} v_1^{\mathrm{Vk}})) = \mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}}\frac{b}{v_0^{\mathrm{Vk}}}\frac{c}{v_1^{\mathrm{Vk}}}(v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} v_0^{\mathrm{Vk}} +_{\mathrm{Vk}} v_0^{\mathrm{Sk}} \cdot_{\mathrm{Vk}} v_1^{\mathrm{Vk}})$

gdw. für alle $a \in \mathbb{K}$ gilt: für alle $b \in V$ gilt: für alle $c \in V$ gilt:
$\mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}}\frac{b}{v_0^{\mathrm{Vk}}}\frac{c}{v_1^{\mathrm{Vk}}}(v_0^{\mathrm{Sk}}) \cdot_V (\mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}}\frac{b}{v_0^{\mathrm{Vk}}}\frac{c}{v_1^{\mathrm{Vk}}}(v_0^{\mathrm{Vk}}) +_V \mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}}\frac{b}{v_0^{\mathrm{Vk}}}\frac{c}{v_1^{\mathrm{Vk}}}(v_1^{\mathrm{Vk}})) =$
$= \mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}}\frac{b}{v_0^{\mathrm{Vk}}}\frac{c}{v_1^{\mathrm{Vk}}}(v_0^{\mathrm{Sk}}) \cdot_V \mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}}\frac{b}{v_0^{\mathrm{Vk}}}\frac{c}{v_1^{\mathrm{Vk}}}(v_0^{\mathrm{Vk}}) +_V \mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}}\frac{b}{v_0^{\mathrm{Vk}}}\frac{c}{v_1^{\mathrm{Vk}}}(v_0^{\mathrm{Sk}}) \cdot_V \mathfrak{V}\frac{a}{v_0^{\mathrm{Sk}}}\frac{b}{v_0^{\mathrm{Vk}}}\frac{c}{v_1^{\mathrm{Vk}}}(v_1^{\mathrm{Vk}})$

gdw. für alle $a \in \mathbb{K}$ und alle $b, c \in V$ gilt:
$a \cdot_V (b +_V c) = a \cdot_V b +_V a \cdot_V c$.

4.3 Allgemeingültige Aussagen

Es gibt enge Zusammenhänge zwischen der syntaktischen Form von Aussagen und ihrer Gültigkeit in Modellen. Tatsächlich lassen sich Beweise vollkommen *formal* auf der Ebene der Syntax führen. Wir beginnen mit einfachen Überlegungen:

Definition 4.7. *Sei σ eine Signatur und $\varphi \in L^\sigma$ eine Aussage.*

a) φ *ist* ***allgemeingültig****, wenn jedes σ-Modell ein Modell von φ ist;*

b) φ *ist* ***erfüllbar****, wenn es ein σ-Modell von φ gibt.*

Satz 4.8. *Folgende Aussagen sind allgemeingültig für alle Terme $t, t_0, t_1, t_2 \in T^\sigma$ und alle Aussagen $\varphi, \psi, \chi \in L^\sigma$.*

a) $t = t$

b) $(t_0 = t_1 \to t_1 = t_0)$

c) $((t_0 = t_1 \wedge t_1 = t_2) \to t_0 = t_2)$

d) $(\varphi \to (\psi \to (\varphi \wedge \psi))$

e) $((\varphi \wedge \psi) \to \varphi)$ *und* $((\varphi \wedge \psi) \to \psi)$

f) $(\varphi \to (\varphi \vee \psi))$ *und* $(\psi \to (\varphi \vee \psi))$

g) $((\varphi \to \chi) \to ((\psi \to \chi) \to ((\varphi \vee \psi) \to \chi)))$

h) $((\neg\varphi \to \neg v_0 = v_0) \to \varphi)$

Beweis. Betrachte ein σ-Modell $\mathfrak{M} = (\mathfrak{A}, \beta)$.
b) Angenommen $\mathfrak{M} \vDash t_0 = t_1$. Dann ist $\mathfrak{M}(t_0) = \mathfrak{M}(t_1)$. Wegen der Symmetrie der Gleichheit ist $\mathfrak{M}(t_1) = \mathfrak{M}(t_0)$. Also ist $\mathfrak{M} \vDash t_1 = t_0$. Danach gilt: $\mathfrak{M} \vDash t_0 = t_1$ impliziert $\mathfrak{M} \vDash t_1 = t_0$, und $\mathfrak{M} \vDash (t_0 = t_1 \to t_1 = t_0)$.
h) Angenommen $\mathfrak{M} \vDash (\neg\varphi \to \neg v_0 = v_0)$. Angenommen $\mathfrak{M} \vDash \neg\varphi$. Dann ist $\mathfrak{M} \vDash \neg$ $v_0 = v_0$. Also ist $\beta(v_0) \neq \beta(v_0)$, Widerspruch. Danach ist $\mathfrak{M} \vDash \neg\varphi$ nicht möglich, und wir erhalten $\mathfrak{M} \vDash \varphi$. Danach gilt: $\mathfrak{M} \vDash (\neg\varphi \to \neg v_0 = v_0)$ impliziert $\mathfrak{M} \vDash \varphi$, und $\mathfrak{M} \vDash ((\neg\varphi \to \neg v_0 = v_0) \to \varphi)$. □

Man beachte, dass dieser Satz über die Allgemeingültigkeit von Aussagen nur die syntaktische Gestalt der Aussagen betrachtet, unabhängig von der Bedeutung von t_0 oder φ. Die bewiesenen Allgemeingültigkeiten entsprechen auch Schritten in (detaillierten) Beweisen. Man kann z.B. d) lesen als: *wenn φ bewiesen ist und ψ bewiesen ist, dann kann man $(\varphi \wedge \psi)$ beweisen.* In diesem Sinne entsprechen e)-h) den Beweismethoden durch *Konjunktionselimination*, *Disjunktion*, *Fallunterscheidung* bzw. *Widerspruch*. Diese Beweismethoden erscheinen auch in den *formalen Beweisen* des nächsten Kapitels.

Kapitel 5
Aussagenlogische formale Beweise

In ausführlichen Beweisen besitzen viele Beweisschritte einen formalen, rechnerischen Charakter. Der Schritt von den Voraussetzungen φ und ψ zu der Konjunktion $(\varphi \wedge \psi)$ ist vollkommen unabhängig von der Bedeutung von φ und ψ; die Schlussfolgerung $(\varphi \wedge \psi)$ lässt sich durch *syntaktisches Operieren* mit den Zeichenreihen φ und ψ bilden. Dies motiviert die Erwartung, dass ein vollkommen ausführlicher Beweis aus einer Folge von Aussagen besteht, die sich jeweils aus früheren Aussagen mit Hilfe syntaktischer Rechenoperationen ergeben.

Wir wollen dieses für den aussagenlogischen Teil unserer Logik demonstrieren. Wir definieren formale Beweise als *Texte* bestimmter Gestalt. Die Grammatik eines korrekten Beweises ist so einfach, dass sie automatisch auf Richtigkeit geprüft werden kann (*proof checking*). Das vorgestellte System entspricht der Software `tutch` (*tutorial proof checker*), die im Internet erhältlich ist: `http://www.tcs.informatik.uni-muenchen.de/~abel/tutch/`

Wir betrachten ein Beispiel, in dem die Tautologie (= allgemeingültige Aussage) $((\varphi \wedge (\varphi \to \psi)) \to \psi)$ bewiesen wird:

Behauptung: (modus ponens) $((\varphi \wedge (\varphi \to \psi)) \to \psi)$
Beweis:
$[(\varphi \wedge (\varphi \to \psi));$
$\varphi;$
$(\varphi \to \psi);$
$\psi];$
$((\varphi \wedge (\varphi \to \psi)) \to \psi)$
qed

Dieser Beweis besteht aus einer Abfolge von Aussagen, die durch Trennzeichen (: [;]) und Schlüsselwörter (*Behauptung*, *Beweis*, *qed*) strukturiert ist. In dem eigentlichen Argument werden Aussagen aus früheren Aussagen nach bestimmten Regeln generiert. Z.B. ergeben sich die Aussagen φ und $(\varphi \wedge \psi)$ durch *Elimination* der Konjunktion $\wedge$ aus der Aussage $(\varphi \wedge (\varphi \to \psi))$.

Der Beweis enthält weiterhin einen *Rahmen* (*frame*)

$$[(\varphi \wedge (\varphi \to \psi)); \varphi; (\varphi \to \psi); \psi].$$

Die erste Aussage $(\varphi \wedge (\varphi \to \psi))$ nach Öffnen des Rahmens durch [ist die *Annahme* des Rahmens; die weiteren Aussagen ergeben sich aus dieser Annahme (und eventuellen früheren Annahmen) auf Grund der Ableitungsregeln. Nach Schließen des Rahmens durch das Symbol] kann aus diesem Rahmen die Implikation

$$((\varphi \wedge (\varphi \to \psi)) \to \psi)$$

geschlossen werden. Allgemeiner gilt: aus dem Rahmen

$$[\varphi_0; \varphi_1; ...; \varphi_{r-1}]$$

kann auf

$$(\varphi_0 \to \varphi_{r-1})$$

geschlossen werden.

Darüber hinaus können weitere Regeln zur Gewinnung von Aussagen angewendet werden, mit denen aussagenlogische Verknüpfungen eingeführt oder eliminiert werden. In dem Beispiel wurde aus den Aussagen φ und $(\varphi \to \psi)$ auf ψ geschlossen.

5.1 Mathematische Texte

Definition 5.1. *Sei σ eine Signatur mit zugehöriger Sprache L^σ.*

a) *Eine Zeichenreihe z ist eine **(Beweis-)Zeile**, wenn es eine Aussage $\varphi \in L^\sigma$ gibt mit*

$$z = \varphi \text{ oder } z = [\varphi \text{ oder } z = \varphi].$$

[φ bezeichnet die Einführung der Annahme φ und kann als „angenommen φ" gelesen werden; $\varphi]$ bezeichnet den Abschluss eines Rahmens zum Beweis von φ und kann als „also φ" gelesen werden.

b) *Ein **(Beweis-)Text** ist eine endliche Folge $z_0, ..., z_{m-1}$ von Beweis-Zeilen.*

c) *Zu einem Text $z_0, ..., z_{m-1}$ definiere rekursiv eine Folge $k(0), k(1), ..., k(m-1)$ von Zeigern auf den Anfang des jeweilig aktiven Rahmens:*
$k(0) = -1$;
seien $k(0), ..., k(i)$ definiert und $i+1 < m-1$; dann sei

$$k(i+1) = \begin{cases} k(i), \text{ falls } z_i = \varphi \in L^\sigma \\ i, \text{ falls } z_i = [\varphi \\ k(k(i)), \text{ falls } z_i = \varphi] \end{cases}$$

*Der Text z_0, ..., z_{m-1} ist **wohlstrukturiert**, wenn die Folge $k(0)$, $k(1)$, ..., $k(m-1)$ definiert ist; die Folge ist genau dann nicht definiert, wenn es ein $z_i = \varphi]$ gibt mit $k(i) = -1$, d.h., wenn] erscheint, ohne dass ein Rahmen geöffnet ist.*

d) *Zu einem wohlstrukturierten Text z_0, ..., z_{m-1} definiere rekursiv die Folge $V_0, V_1, ..., V_{m-1}$ der **lokalen Voraussetzungen**:*
$V_0 = \emptyset$;
seien $V_0, ..., V_i$ definiert und $i+1 < m-1$; dann sei

$$V_{i+1} = \begin{cases} V_i \cup \{z_i\}, \text{ falls } z_i = \varphi \in L^\sigma \\ V_i \cup \{\varphi\}, \text{ falls } z_i = [\varphi \\ V_{k(i)} \cup \{(\psi \to \varphi)\}, \text{ falls } z_i = \varphi] \text{ und } z_{k(i)} = [\psi \end{cases}$$

Wir demonstrieren diese Definitionen an dem Text

$$\begin{array}{ll} 0: & [\varphi \\ 1: & [\psi \\ 2: & \varphi] \\ 3: & (\psi \rightarrow \varphi)] \\ 4: & (\varphi \rightarrow (\psi \rightarrow \varphi)) \end{array}$$

in dem die Tautologie $(\varphi \rightarrow (\psi \rightarrow \varphi))$ bewiesen wird:

i	z_i	$k(i)$	V_i
0:	$[\varphi$	-1	$\emptyset$
1:	$[\psi$	0	$\{\varphi\}$
2:	$\varphi]$	1	$\{\varphi, \psi\}$
3:	$(\psi \rightarrow \varphi)]$	0	$\{\varphi, (\psi \rightarrow \varphi)\}$
4:	$(\varphi \rightarrow (\psi \rightarrow \varphi))$	-1	$\{(\varphi \rightarrow (\psi \rightarrow \varphi))\}$

5.2 Formale Beweise mit $\wedge$, $\rightarrow$ und $\leftrightarrow$

In einem formalen Beweis wird aus den Voraussetzungen V_i mit Hilfe von *Schlussregeln* die nächste Zeile z_i gebildet. Wir diskutieren verschiedene Schlussregeln und entsprechend gebildete Beweise.

Schlussregeln werden in der Form

$$\frac{\chi_0 \ \chi_1 \ \cdots \ \chi_{r-1}}{\psi}$$

notiert: aus den Voraussetzungen χ_0, χ_1, ..., χ_{r-1} kann auf ψ geschlossen werden. Dieses Schließen ist zunächst rein formal. Wir werden aber nur solche Regeln betrachten, die dem üblichen logischen Schließen entsprechen. Die Regeln dienen zur Einführung oder Elimination von logischen Symbolen in Aussagen. Wir betrachten zunächst Regeln, die die aussagenlogischen Verknüpfungen $\wedge$ und $\rightarrow$ betreffen.

$\wedge$-Regeln:

			tutch − Bezeichnung
$\wedge$-Einführung:	$\wedge I$	$\dfrac{\varphi \quad \psi}{(\varphi \wedge \psi)}$	`AndI`
$\wedge$-Elimination:	$\wedge E$	$\dfrac{(\varphi \wedge \psi)}{\varphi}$	`AndEL`
$\wedge$-Elimination:	$\wedge E$	$\dfrac{(\varphi \wedge \psi)}{\psi}$	`AndER`

$\rightarrow$-Regeln:

			tutch − Bezeichnung
$\rightarrow$-Elimination:	$\rightarrow E$	$\dfrac{\varphi \ (\varphi \rightarrow \psi)}{\psi}$	`ImpI`

$\leftrightarrow$-Regel:

Die Aussage $(\varphi \leftrightarrow \psi)$ wird als *Abkürzung* für $((\varphi \to \psi) \wedge (\psi \to \varphi))$ benutzt. Damit wird $\leftrightarrow$ durch die Regeln für $\wedge$ und $\to$ erfasst.

Definition 5.2. *Sei $R_0, ..., R_{n-1}$ eine Liste von Regeln. Ein Beweis-Text $z_0, ..., z_{m-1}$ ist ein **(formaler) Beweis** entsprechend den Regeln $R_0, ..., R_{n-1}$ wenn:*

a) $z_0, ..., z_{m-1}$ ist wohlstrukturiert; es seien $k(0), ..., k(m-1)$ und $V_0, ..., V_{m-1}$ die entsprechenden Zeiger und Voraussetzungsmengen;

b) für alle Beweis-Zeilen $z_i = \psi_i$ oder $z_i = \psi_i]$ mit $\psi_i \in L^\sigma$ ist $\psi_i \in V_i$ („Kopieren einer Voraussetzung"), oder es gibt eine Regel

$$\frac{\chi_0 \ \chi_1 \ \cdots \ \chi_{r-1}}{\psi}$$

aus der Liste der Regeln und Aussagen $\varphi_0, ..., \varphi_{r-1} \in V_i$, so dass sich ψ_i aus $\varphi_0, ..., \varphi_{r-1}$ mit Hilfe der Regel ergibt.

*Der Beweis ist ein **Beweis** der Aussage ψ, wenn $\psi = z_{m-1}$ die letzte Zeile des Beweises ist und $k(m-1) = -1$.*

Wir betrachten Beispiele von formalen Beweisen entsprechend den bisher eingeführten Regeln und Versionen der gleichen Beweise im System `tutch`.

1. Ein formaler Beweis der Tautologie $(\varphi \to (\psi \to \varphi))$:

i	z_i	Kommentar
0:	$[\varphi$	Einführung einer Annahme
1:	$[\psi$	Einführung einer Annahme
2:	$\varphi]$	Kopieren einer Voraussetzung
3:	$(\psi \to \varphi)]$	Kopieren einer Voraussetzung
4:	$(\varphi \to (\psi \to \varphi))$	Kopieren einer Voraussetzung

Dieser Text wird, abgesehen von einfachen Formatierungskonventionen, vom System `tutch` akzeptiert und mit den den benutzten Ableitungsregeln ergänzt. Als Eingabe verwenden wir die Datei `Implikation1.tut`

```
proof Implikation1: (Phi => (Psi => Phi)) =
begin
[Phi;
[Psi;
Phi];
(Psi => Phi)];
(Phi => (Psi => Phi))
end;
```

```
   Shell session inside TeXmacs
shell] cd V/Logik_und_diskrete_Strukturen/tutch
shell] tutch
   TUTCH 0.52 beta, $Date: 2002/10/24 19:25:49 $
   Usage: tutch [options] files
   Options:
```

```
    -r file  Checks files against requirements in 'file'
    -q | -Q  Be quiet | really quiet
    -v | -V  Be verbose | maximum verbose
    -h       Print this help message
  Default extensions:
    .tut     for proof files
    .req     for requirement files
  Any file without extension will be given the proper extension
  automatically.
  If no files are given, but a requirements file 'file[.req]' is
  specified,
  'file.tut' will be checked against 'file.req'.
  Warning: This command DOES NOT SUBMIT your files!

shell] tutch -V Implikation1
  TUTCH 0.52 beta, $Date: 2002/10/24 19:25:49 $
  [Opening file Implikation1.tut]

  Proving Implikation1: Phi => Psi => Phi ... (classically)
    1  [ Phi;
    2    [ Psi;
    3      Phi ];                by Hyp 1
    4    Psi => Phi ];           by ImpI 3
    5  Phi => Psi => Phi         by ImpI 4
  QED
  [Closing file Implikation1.tut]

shell]
```

2. Ein formaler Beweis der Tautologie $(\varphi \rightarrow (\psi \rightarrow (\varphi \wedge \psi)))$:

i	z_i	Kommentar
0:	$[\varphi$	Einführung einer Annahme
1:	$[\psi$	Einführung einer Annahme
2:	$(\varphi \wedge \psi)]$	$\wedge$-Einführung mit 0 und 1
3:	$(\psi \rightarrow (\varphi \wedge \psi))]$	Kopieren einer Voraussetzung
4:	$(\varphi \rightarrow (\psi \rightarrow (\varphi \wedge \psi)))$	Kopieren einer Voraussetzung

```
proof Und_Einfuehrung: (Phi => (Psi => (Phi & Psi))) =
begin
[Phi;
[Psi;
(Phi & Psi)];
(Psi => (Phi & Psi))];
(Phi => (Psi => (Phi & Psi)))
end;
```

```
shell] tutch -V Und_Einfuehrung
   TUTCH 0.52 beta, $Date: 2002/10/24 19:25:49 $
   [Opening file Und_Einfuehrung.tut]

   Proving Und_Einfuehrung: Phi => Psi => Phi & Psi ...
     1  [ Phi;
     2    [ Psi;
     3      Phi & Psi ];                  by AndI 1 2
     4    Psi => Phi & Psi ];             by ImpI 3
     5  Phi => Psi => Phi & Psi           by ImpI 4
   QED
   [Closing file Und_Einfuehrung.tut]

shell]
```

3. Ein formaler Beweis der Tautologie $((\varphi \wedge \psi) \rightarrow (\psi \wedge \varphi))$:

i	z_i	Kommentar
0:	$[(\varphi \wedge \psi)$	Einführung einer Annahme
1:	φ	$\wedge$-Elimination mit 0
2:	ψ	$\wedge$-Elimination mit 0
3:	$(\psi \wedge \varphi)]$	$\wedge$-Einführung mit 2 und 1
4:	$((\varphi \wedge \psi) \rightarrow (\psi \wedge \varphi))$	Kopieren einer Voraussetzung

```
proof Und_Kommutativitaet: ((Phi & Psi) => (Psi & Phi)) =
begin
[ (Phi & Psi);
Phi;
Psi;
(Psi & Phi)];
((Phi & Psi) => (Psi & Phi))
end;
```

```
shell] tutch -V Und_Kommutativitaet
   TUTCH 0.52 beta, $Date: 2002/10/24 19:25:49 $
   [Opening file Und_Kommutativitaet.tut]

   Proving Und_Kommutativitaet: Phi & Psi => Psi & Phi ...
     1  [ Phi & Psi;
     2    Phi;                            by AndEL 1
     3    Psi;                            by AndER 1
     4    Psi & Phi ];                    by AndI 3 2
     5  Phi & Psi => Psi & Phi            by ImpI 4
   QED
   [Closing file Und_Kommutativitaet.tut]

shell]
```

4. Ein formaler Beweis der Tautologie $((\varphi \wedge \varphi) \leftrightarrow \varphi)$.

i	z_i	Kommentar
0:	$[(\varphi \wedge \varphi)$	Einführung einer Annahme
1:	$\varphi]$	$\wedge$-Elimination mit 0
2:	$((\varphi \wedge \varphi) \rightarrow \varphi)$	Kopieren einer Voraussetzung
3:	$[\varphi$	Einführung einer Annahme
4:	$(\varphi \wedge \varphi)]$	$\wedge$-Einführung mit 3 und 3
5:	$(\varphi \rightarrow (\varphi \wedge \varphi))$	Kopieren einer Voraussetzung
6:	$((\varphi \wedge \varphi) \leftrightarrow \varphi)$	$\wedge$-Einführung mit 2 und 5

```
   Shell session inside TeXmacs
shell] tutch -V Und_Idempotenz
   TUTCH 0.52 beta, $Date: 2002/10/24 19:25:49 $
   [Opening file Und_Idempotenz.tut]

   Proving Und_Idempotenz: Phi & Phi <=> Phi ...
     1  [ Phi & Phi;
     2    Phi ];                    by AndEL 1
     3  Phi & Phi => Phi;           by ImpI 2
     4  [ Phi;
     5    Phi & Phi ];              by AndI 4 4
     6  Phi => Phi & Phi;           by ImpI 5
     7  Phi & Phi <=> Phi           by AndI 3 6
   QED
   [Closing file Und_Idempotenz.tut]

shell]
```

5.3 Beweisregeln für $\vee$ und $\neg$

Wir führen Regeln für die übrigen aussagenlogischen Verknüpfungen ein, die auch im System `tutch` implementiert sind. Entsprechend der in `tutch` implementierten Logik benutzen wir zusätzliche Aussagenkonstanten $\top$ (Top, `T` in `tutch`) für „wahr" und $\bot$ (Bottom, `F` in `tutch`) für „falsch".

$\vee$-Regeln:

			`tutch` − Bezeichnung
$\vee$-Einführung:	$\vee I$	$\dfrac{\varphi}{(\varphi \vee \psi)}$	`OrI1`
$\vee$-Einführung:	$\vee I$	$\dfrac{\psi}{(\varphi \vee \psi)}$	`OrI2`
Fallunterscheidung ($\vee$-Elimination):	$\vee E$	$\dfrac{(\varphi \vee \psi)\ (\varphi \rightarrow \chi)\ (\psi \rightarrow \chi)}{\chi}$	`OrE`

Konstanten-Regeln:

			`tutch − Bezeichnung`
$\top$-Einführung:	$\top I$	$\dfrac{}{\top}$	`TrueI`
$\bot$-Elimination:	$\bot E$	$\dfrac{\bot}{\varphi}$	`FalseE`

¬-Regel:

Die Aussage $\neg\varphi$ wird als *Abkürzung* für $\varphi \rightarrow \bot$ benutzt.

			`tutch − Bezeichnung`
Widerspruchsregel:	Wid	$\dfrac{(\neg\varphi \rightarrow \bot)}{\varphi}$	`Class`

1. Ein formaler Beweis der Tautologie $((\neg\varphi \vee \psi) \leftrightarrow (\varphi \rightarrow \psi))$:

i	z_i	Kommentar
0:	$[(\neg\varphi \vee \psi)$	Einführung einer Annahme
1:	$[\varphi$	Einführung einer Annahme
2:	$[\neg\varphi$	Einführung einer Annahme
3:	$\bot$	$\rightarrow$-Elimination mit 1 und 2
4:	$\psi]$	$\bot$-Elimination mit 3
5:	$(\neg\varphi \rightarrow \psi)$	Kopieren einer Voraussetzung
6:	$[\psi$	Einführung einer Annahme
7:	$\psi]$	Kopieren einer Voraussetzung
8:	$(\psi \rightarrow \psi)$	Kopieren einer Voraussetzung
9:	$\psi]$	Fallunterscheidung mit 0,5 und 8
10:	$(\varphi \rightarrow \psi)]$	Kopieren einer Voraussetzung
11:	$((\neg\varphi \vee \psi) \rightarrow (\varphi \rightarrow \psi))$	Kopieren einer Voraussetzung
12:	$[(\varphi \rightarrow \psi)$	Einführung einer Annahme
13:	$[\neg(\neg\varphi \vee \psi)$	Einführung einer Annahme
14:	$[\neg\varphi$	Einführung einer Annahme
15:	$(\neg\varphi \vee \psi)$	$\vee$-Einführung mit 14
16:	$\bot]$	$\rightarrow$-Elimination mit 13 und 15
17:	φ	Widerspruchsregel mit 14 und 16
18:	ψ	$\rightarrow$-Elimination mit 12 und 17
19:	$(\neg\varphi \vee \psi)$	$\vee$-Einführung mit 18
20:	$\bot]$	$\rightarrow$-Elimination mit 13 und 19
21:	$(\neg\varphi \vee \psi)]$	Widerspruchsregel mit 13 und 20
22:	$((\varphi \rightarrow \psi) \rightarrow (\neg\varphi \vee \psi))$	Kopieren einer Voraussetzung
23:	$((\neg\varphi \vee \psi) \rightarrow (\varphi \rightarrow \psi))$	$\wedge$-Einführung mit 11 und 22

```
shell] tutch -V Implikation
   TUTCH 0.52 beta, $Date: 2002/10/24 19:25:49 $
```

```
[Opening file Implikation.tut]

Proving impDef: ~Phi | Psi <=> Phi => Psi ... (classically)
  1  [ ~Phi | Psi;
  2    [ Phi;
  3      [ ~Phi;
  4        F;                          by ImpE 3 2
  5        Psi ];                      by FalseE 4
  6      [ Psi;
  7        Psi ];                      by Hyp 6
  8      Psi ];                        by OrE 1 5 7
  9    Phi => Psi ];                   by ImpI 8
 10  ~Phi | Psi => Phi => Psi;         by ImpI 9
 11  [ Phi => Psi;
 12    [ ~(~Phi | Psi);
 13      [ ~Phi;
 14        ~Phi | Psi;                 by OrIL 13
 15        F ];                        by ImpE 12 14
 16      Phi;                          by Class 15
 17      Psi;                          by ImpE 11 16
 18      ~Phi | Psi;                   by OrIR 17
 19      F ];                          by ImpE 12 18
 20    ~Phi | Psi ];                   by Class 19
 21  (Phi => Psi) => ~Phi | Psi;       by ImpI 20
 22  ~Phi | Psi <=> Phi => Psi         by AndI 10 21
QED
[Closing file Implikation.tut]

shell]
```

5.4 Beweisansätze

Die Methoden für das formale Beweisen liefern auch Ansätze für das gewöhnliche Beweisen. Um eine Aussage bestimmter Gestalt zu beweisen, erlauben die formalen Regeln, die die Beweisaufgabe in ,,kleinere" Beweisaufgaben zerlegen.

- um eine Konjunktion $(\varphi \wedge \psi)$ zu beweisen, kann man Beweise von φ und von ψ führen.
- um eine Implikation $(\varphi \to \psi)$ zu beweisen, kann man φ annehmen und ψ unter dieser (zusätzlichen) Annahme beweisen.
- um eine Disjunktion $(\varphi \vee \psi)$ zu beweisen, kann man φ oder ψ beweisen.
- um eine Aussage φ zu beweisen, kann man einen Widerspruchsbeweis führen, bei dem $\neg\varphi$ angenommen wird und daraus ein Widerspruch hergeleitet wird.

- um eine Aussage χ zu beweisen kann man mit einer Fallunterscheidung vorgehen: man zeigt Implikation $(\varphi_0 \rightarrow \chi)$, ..., $(\varphi_{m-1} \rightarrow \chi)$ und eine Disjunktion $(\varphi_0 \vee ... \vee \varphi_{m-1})$.
- eine einfache Form der Fallunterscheidung ist die Situation $\varphi_0 = \varphi$ und $\varphi_1 = \neg\varphi$: es genügt, die Implikationen $(\varphi \rightarrow \chi)$ und $(\neg\varphi \rightarrow \chi)$ zu zeigen.

Regeln dieser Art liefern einen ersten Anhaltspunkt für das Vorgehen (im aussagenlogischen Fall). Grundsätzlich aber ist Finden von Beweisen von hoher Komplexität ist (Rechenzeit), es gibt es beweisbare Aussagen, deren Beweis nicht durch einfache Regeln gefunden werden kann. Beispielsweise gibt es *unendlich* viele Möglichkeiten, wie bei einem Beweis durch Fallunterscheidung die Fallunterscheidung $\varphi / \neg\varphi$ gewählt werden kann.

Kapitel 6
Quantorenlogische formale Beweise

6.1 Allaussagen

Wir beginnen mit allgemeinen Überlegungen zum Beweis von Quantoren-Aussagen. Eine All-Aussage $\forall x\,\varphi$ kann für unendliche Strukturen nicht dadurch bewiesen werden, dass man φ für unendlich viele Elemente einzeln überprüft. Das Vorgehen besteht vielmehr darin, ein ,,allgemeines" Element der Struktur zu betrachten und φ hierfür nachzuweisen.

Möchte man z.B. für Vektorräume zeigen, dass

$$\forall x\ -x = (-1)\cdot x$$

ist, so betrachte man einen ,,beliebigen" Vektor y und weise für diesen nach, dass $-y = (-1)\cdot y$. Dass y beliebig ist, bedeutet, dass die Variable y zu Anfang dieses Argumentes nicht frei in den gerade gültigen Voraussetzungen vorkommt. Einzig der Typ der Variable wird festgelegt:

...
[Betrachte einen Vektor y
...
$-y = (-1)\cdot y$]
(Also gilt) $\forall x\ -x = (-1)\cdot x$
...

Dies ist eine Beweismethode, die wir in die formalen Beweise aufnehmen. Beim Schließen eines Rahmens wird eine All-Aussage zu den lokalen Voraussetzungen hinzugefügt. Wir ändern die Definition 5.1 dementsprechend ab:

Definition 6.1. *Sei $z_0, ..., z_{m-1}$ ein wohlstrukturierter Text mit Zeigern $k(0), ..., k(m-1)$. Dann definiere die modifizierte Folge $V_0, V_1, ..., V_{m-1}$ der* ***lokalen Voraussetzungen****:*
$V_0 = \emptyset$*;*
seien $V_0, ..., V_i$ definiert und $i+1 < m-1$; dann sei

$$V_{i+1} = \begin{cases} V_i \cup \{z_i\}, \text{ falls } z_i = \varphi \in L^\sigma \\ V_i \cup \{\varphi\}, \text{ falls } z_i = [\varphi \\ V_{k(i)} \cup \{\forall x_0 ... \forall x_{m-1}(\psi \to \varphi) \mid \{x_0, ..., x_{m-1}\} \subseteq (\mathrm{frei}(\psi \to \varphi) \setminus \mathrm{frei}(V_{k(i)}))\}, \\ \qquad\qquad \text{falls } z_i = \varphi] \text{ und } z_{k(i)} = [\psi \\ V_{k(i)} \cup \{\forall x_0 ... \forall x_{m-1}\varphi) \mid \{x_0, ..., x_{m-1}\} \subseteq (\mathrm{frei}(\varphi) \setminus \mathrm{frei}(V_{k(i)}))\}, \\ \qquad\qquad \text{falls } z_i = \varphi] \text{ und } z_{k(i)} = [\top \end{cases}$$

Auf diese Art können in einen Beweis ∀-Aussagen eingeführt werden. Für die *Elimination* des Allquantors gibt es die folgende Regel:

∀-Regeln:

$$\forall\text{-Elimination: } \forall E \ \frac{\forall x\,\varphi(x)}{\varphi(t)} \qquad \begin{array}{l}\texttt{tutch} - \texttt{Bezeichnung} \\ \texttt{ForallE}\end{array}$$

Wir führen nun einen formalen Beweis, dass die Reihenfolge in Blöcken von ∀-Quantoren nicht relevant ist:

i	z_i	Kommentar
0:	$[\forall x \forall y \varphi(x,y)$	Einführung einer Annahme
1:	$[\top$	Einführung einer trivialen Annahme
2:	$\varphi(x,y)]$	∀-Elimination mit 0
3:	$\forall y \forall x \varphi(x,y)]$	Kopieren einer Voraussetzung
4:	$(\forall x \forall y \varphi(x,y) \rightarrow \forall y \forall x \varphi(x,y))$	Kopieren einer Voraussetzung

In `tutch` gibt es wie in der früher eingeführten formalen Sprache nur Variablen bestimmten Typs. Der entsprechende `tutch`-Text sieht folgendermaßen aus:

```
classical proof QuantorenReihenfolge:
((!x:t.!y:t. Phi(x,y)) (!y:t.!x:t. Phi(x,y))) =
begin
[!x:t.!y:t. Phi(x,y);
 [b:t;
  [a:t;
    !y:t. Phi(a,y);
    Phi(a,b)];
  !x:t. Phi(x,b)];
!y:t.!x:t. Phi(x,y)];
((!x:t.!y:t. Phi(x,y)) (!y:t.!x:t. Phi(x,y)))
end;
```

`tutch` akzeptiert diese Datei:

```
   Shell session inside TeXmacs
shell] cd ~/V/Logik_und_diskrete_Strukturen/tutch
shell] tutch -V Quantoren-Reihenfolge.tut
   TUTCH 0.52 beta, $Date: 2002/10/24 19:25:49 $
   [Opening file Quantoren-Reihenfolge.tut]
```

```
Proving QuantorenReihenfolge: (!x:t. !y:t. Phi (x, y)) => !y:t.
!x:t. Phi (x, y) ... (classically)
  1  [ !x:t. !y:t. Phi (x, y);

  2    [ b: t;

  3      [ a: t;

  4        !y:t. Phi (a, y);
     by ForallE 1 3
  5       Phi (a, b) ];
     by ForallE 4 2
  6      !x:t. Phi (x, b) ];
     by ForallI 5
  7    !y:t. !x:t. Phi (x, y) ];
     by ForallI 6
  8  (!x:t. !y:t. Phi (x, y)) => !y:t. !x:t. Phi (x, y)
     by ImpI 7
QED
[Closing file Quantoren-Reihenfolge.tut]


shell]
```

Es gibt noch viele andere Möglichkeiten, All-Aussagen zu beweisen. Z.B. beweist man mit dem Prinzip der *vollständigen Induktion* All-Aussagen über natürliche Zahlen n, indem man den Induktionsanfang für $n = 0$ beweist und zeigt, dass die Eigenschaft an der Stelle $n+1$ gilt, wenn sie an der Stelle n gilt.

6.2 Existenzaussagen

Eine Existenzaussage $\exists x \varphi(x)$ kann man beweisen, indem man $\varphi(t)$ für einen konkreten Term t nachweist. Hierdurch wird ein Existenzquantor in das Argument eingeführt. Dual hierzu gibt es eine Regel zur Elimination von Existenzquantoren.

∃-Regeln:

			`tutch − Bezeichnung`
∃-Einführung	$\exists I$	$\dfrac{\varphi(t)}{\exists x \varphi(x)}$	`ExistsI`
∃-Elimination:	$\exists E$	$\dfrac{\exists x \varphi(x) \ (\varphi(y) \to \psi)}{\psi}$	`ExistsE`

Bei der $\exists$-Elimination ist als Nebenbedingung zu fordern, dass die Variable y „frei" gewählt ist: y ist nicht frei in den Voraussetzungen, die zum Zeitpunkt der Einführung von $\varphi(y)$ aktiv sind; außerdem kommt y nicht frei in ψ vor. Dies wird deutlicher in dem formalen Beweis

i	z_i	Kommentar
0:	...	...
	$\vdots$	$\vdots$
i:	$\exists x \varphi(x)$	
	$\vdots$	$\vdots$
j:	$[\varphi(y)$	„Wähle y mit der Eigenschaft φ"
	$\vdots$	$\vdots$
k:	$\psi]$	
$k+1$:	ψ	$\exists$-Elimination mit i,j und k
	$\vdots$	$\vdots$

Wir bringen noch ein `tutch`-Beispiel zur Umbenennung von Variablen:

i	z_i	Kommentar
0:	$[\exists x \varphi(x)$	Einführung einer Annahme
1:	$[\varphi(y)$	„Wähle y mit der Eigenschaft φ"
2:	$\exists y \varphi(y)]$	$\exists$-Einführung mit 1
2:	$\exists y \varphi(y)]$	$\forall$-Elimination mit 0,1,2
3:	$(\exists x \varphi(x) \rightarrow \exists y \varphi(y))$	Kopieren einer Voraussetzung

Die zugehörige Datei lautet:

```
proof Umbenennung: ((?x:t. Phi(x)) => (?y:t. Phi(y))) =
begin
[ ?x:t. Phi(x);
  [ y:t, Phi(y);
    ?y:t. Phi(y)];
  ?y:t. Phi(y)];
((?x:t. Phi(x)) => (?y:t. Phi(y)))
end;
```

```
shell] cd ~/V/Logik_und_diskrete_Strukturen/tutch
shell] tutch -V Umbenennung.tut
```

```
   TUTCH 0.52 beta, $Date: 2002/10/24 19:25:49 $
   [Opening file Umbenennung.tut]

   Proving Umbenennung: (?x:t. Phi x) => ?y:t. Phi y ...
     1  [ ?x:t. Phi x;
     2    [ y: t, Phi y;
     3      ?y:t. Phi y ];                          by Hyp 1
     4    ?y:t. Phi y ];                            by Hyp 1
     5  (?x:t. Phi x) => ?y:t. Phi y                by ImpI 4
   QED
```

```
[Closing file Umbenennung.tut]

shell]
```

6.3 Der Gödelsche Vollständigkeitssatz

Wir stellen die eingeführten Regeln formalen Beweisens noch einmal zusammen.

Definition 6.2. *Die* ***Regeln des natürlichen Schließens*** *sind:*

∧-Regeln:

			tutch − Bezeichnung
∧-*Einführung*:	$\wedge I$	$\dfrac{\varphi \quad \psi}{(\varphi \wedge \psi)}$	`AndI`
∧-*Elimination*:	$\wedge E$	$\dfrac{(\varphi \wedge \psi)}{\varphi}$	`AndEL`
∧-*Elimination*:	$\wedge E$	$\dfrac{(\varphi \wedge \psi)}{\psi}$	`AndER`

→-Regeln:

→-*Elimination*:	$\rightarrow E$	$\dfrac{\varphi \ \ (\varphi \rightarrow \psi)}{\psi}$	`ImpI`

∨-Regeln:

∨-*Einführung*:	$\vee I$	$\dfrac{\varphi}{(\varphi \vee \psi)}$	`OrI1`
∨-*Einführung*:	$\vee I$	$\dfrac{\psi}{(\varphi \vee \psi)}$	`OrI2`
Fallunterscheidung (∨-*Elimination*):	$\vee E$	$\dfrac{(\varphi \vee \psi) \ (\varphi \rightarrow \chi) \ (\psi \rightarrow \chi)}{\chi}$	`OrE`

Konstanten-Regeln:

⊤-*Einführung*:	$\top I$	$\dfrac{}{\top}$	`TrueI`
⊥-*Elimination:*	$\bot E$	$\dfrac{\bot}{\varphi}$	`FalseE`

¬-Regel:

Die Aussage $\neg\varphi$ *wird als Abkürzung für* $\varphi \rightarrow \bot$ *benutzt.*

Widerspruchsregel:	Wid	$\dfrac{(\neg\varphi \rightarrow \bot)}{\varphi}$	`Class`

∀-Regeln:

∀-*Elimination*:	$\forall E$	$\dfrac{\forall x\, \varphi(x)}{\varphi(t)}$	`ForallE`

***∃-Regeln**:*

$$\exists\text{-}\textit{Einführung} \quad \exists I \quad \frac{\varphi(t)}{\exists x\,\varphi(x)} \qquad \texttt{ExistsI}$$

$$\exists\text{-}\textit{Elimination}: \quad \exists E \quad \frac{\exists x\,\varphi(x) \;\; (\varphi(y) \to \psi)}{\psi} \qquad \texttt{ExistsE}$$

Definition 6.3. *Eine Aussage φ ist **formal beweisbar**, wenn es eine formalen Beweis im Sinne der Definition 5.2 mit den Regeln des natürlichen Schließens gibt.*

Wir haben exemplarisch gesehen, dass sich die üblichen mathematischen Argumentationen auf diese Regeln zurückführen lassen. Erstaunlicherweise kann man das auch mathematisch *beweisen*:

Satz 6.4. ***(Gödelscher Vollständigkeitssatz).** Eine Aussage φ ist genau dann allgemeingültig, wenn sie formal beweisbar ist.*

Bemerkung 6.5. Der GÖDELsche Vollständigkeitssatz ist der Hauptsatz der mathematischen Logik. Er verbindet auf bestmögliche Weise Semantik und Syntax formaler Sprachen. Dass jede formal beweisbare Aussage allgemeingültig ist, entspricht der *Korrektheit* der angegebenen formalen Beweismethode. Selbstverständlich nimmt man nur solche Schlussregeln auf, die korrekt sind.

Dass umgekehrt jede allgemeingültige Aussage formal bewiesen werden kann, ist schwieriger zu zeigen und ist das eigentliche GÖDELsche Theorem.

Der Gödelsche Vollständigkeitssatz hat über die mathematische Logik hinaus viele Folgerungen, die auf der Verbindung *natürliche Sprache - formale Sprache - formale Semantik - allgemeine Semantik* beruhen. Wir erwähnen einige Bereiche.

Bemerkung 6.6. Der Vollständigkeitssatz liefert ein *absolutes Korrektheitskriterium* für Beweise. Ein mathematischer Beweis ist genau dann korrekt, wenn er (im Prinzip) in einen formalen Beweis umgeschrieben werden kann. Obwohl man für gewöhnlich informell argumentiert, kann man in Zweifelsfällen Argumente soweit formalisieren und in kleinste Zwischenschritte unterteilen, dass sie rein formal und auch von einem Computer überprüft werden können.

Bemerkung 6.7. Der Erfolg der formalen Methode in der Mathematik regt auch andere Bereiche an, ihre Aussagen und Erkenntnismethoden nach Möglichkeit zu formalisieren. Dies geht einher mit der Erfassung der Welt als Daten, die mit Algorithmen verarbeitet werden.

Bemerkung 6.8. *Automatische Beweisen.* Im Prinzip können alle allgemeingültigen Aussagen φ *automatisch* bewiesen werden: zähle alle Texte auf und überprüfe, ob sie ein formaler Beweis von φ sind. Nach dem GÖDELschen Vollständigkeitssatz hat φ einen formalen Beweis, der durch dieses Vorgehen schließlich gefunden wird. Allerdings ist dieses Vorgehen aus *Komplexitätsgründen* im Allgemeinen praktisch nicht realisierbar. Für eingeschränkte Bereiche gibt es inzwischen automatische Beweiser.

Bemerkung 6.9. *Logisches Programmieren.* Ein Spezialfall des automatischen Beweisens ist das Logische Programmieren (`Prolog`). Programme bestehen aus Quantorenlogischen Aussagen (beschränkter Komplexität), die Ausführung des Programms besteht in der systematischen Anwendung von Schlussregeln auf das Programm.

Bemerkung 6.10. *Künstliche Intelligenz.* Ein erster Ansatz zur Realisierung von künstlicher Intelligenz bestand in der Formulierung von Umwelttatsachen und Fragen in der formalen Logik. Mit den Methoden des automatischen Beweisens wurde dann nach einem Beweis der Frage oder ihrer Negation gesucht. Diese Ansätze haben sich wegen der erwähnten Komplexitätsprobleme als unrealistisch herausgestellt.

Bemerkung 6.11. *Hoare Logik.* Angeregt von dem Erfolg der Quantorenlogik wurden angepasste Logiken für viele Bereiche entwickelt. Von besonderer Wichtigkeit für die Informatik sind Logiken, mit denen sich die Semantik von Algorithmen beschreiben lässt. Im Gegensatz zu den Strukturen der Quantorenlogik sind Algorithmen dynamisch, jede Anweisung hat einen Zustand vor der Ausführung und einen (veränderten) Zustand nach der Ausführung. Die Hoare-Logik beschreibt solche Übergänge, und sie besitzt Schlussregeln, mit denen man die Korrektheit von Programmen beweisen kann.

Bemerkung 6.12. *Gödelscher Unvollständigkeitssatz.* Der Vollständigkeitssatz darf nicht mit dem **Un**vollständigkeitssatz verwechselt werden, der mehr öffentliche Aufmerksamkeit findet.

Die mathematische Logik hat trotz der (prinzipiellen) Universalität ihrer Methode auch ihre Grenzen studiert. In genügend starken Theorien lässt sich sich das bekannte *Lügner-Paradoxon* („alle Kreter sind Lügner") des Epimenides nachbilden, das der umgangssprachlichen Aussage „dieser Satz ist falsch" entspricht. Man kann diesem Lügner-Satz keinen Wahrheitswert geben, sein Wahrheitswert ist nicht entscheidbar. Ähnlich kann man einen zahlentheoretischen Satz angeben, der mit den Peanoschen Axiomen der Zahlentheorie nicht entschieden werden kann. Dies ist ist der Inhalt des (ersten) Gödelschen Unvollständigkeitssatzes.

Der Unvollständigkeitssatz steht in enger Beziehung zum *Halteproblem* der Rekursionstheorie und theoretischen Informatik.

Kapitel 7
Kombinatorik

Die (endliche) Kombinatorik untersucht die Frage: wieviel Elemente enthält eine endliche Menge. Die Anzahl der Elemente wird durch den Grundbegriff der *Kardinalität* gegeben.

Definition 7.1. *Die* ***Mächtigkeit*** *oder* ***Kardinalität*** *einer endlichen Menge S ist die Anzahl der in S enthaltenen Elemente und wird mit $|S|$ bezeichnet. $|S| = m$ genau dann, wenn sich S aufzählen als*

$$S = \{s_0, ..., s_{m-1}\}$$

mit paarweise verschiedenen Elemente $s_0, ..., s_{m-1}$ aufzählen lässt.

Insbesondere ist $|\emptyset| = 0$ und $|\{s\}| = 1$. Wir werden im folgenden verschiedene arithmetische Gesetzmäßigkeiten der $|\ |$-Funktion studieren.

Satz 7.2. *Seien A und B endliche Mengen. Dann gilt:*

a) *Wenn A und B zueinander disjunkt sind, d.h. $A \cap B = \emptyset$, dann ist $|A \cup B| = |A| + |B|$.*

b) *Seien $A_0, ..., A_{m-1}$ endliche Mengen sind, die paarweise disjunkt sind, d.h. für $i < j < m$ ist $A_i \cap A_j = \emptyset$. Dann ist*

$$|A_0 \cup ... \cup A_{m-1}| = |A_0| + ... + |A_{m-1}|.$$

c) $|A \cup B| = |A| + |B| - |A \cap B|$ *(Einschluss-Ausschluss-Prinzip).*

d) $|A \times B| = |A| \cdot |B|$.

e) *Seien $A_0, ..., A_{m-1}$ endliche Mengen. Dann ist*

$$|A_0 \times ... \times A_{m-1}| = |A_0| \cdot ... \cdot |A_{m-1}|.$$

Beweis. a) Seien $A = \{a_0, ..., a_{m-1}\}$ und $B = \{b_0, ..., b_{n-1}\}$ mit jeweils paarweise verschiedenen Elementen $a_0, ..., a_{m-1}$ und $b_0, ..., b_{n-1}$. Da A und B disjunkt sind, sind die Elemente $a_0, ..., a_{m-1}, b_0, ..., b_{n-1}$ paarweise verschieden.

$$A \cup B = \{a_0, ..., a_{m-1}, b_0, ..., b_{n-1}\}$$

und daher

$$|A \cup B| = m + n = |A| + |B|.$$

b) Durch vollständige Induktion über $m \geqslant 1$.
Induktionsanfang: $m = 1$. $|A_0| = |A_0|$ gilt trivial.
Induktionsschritt: Die Behauptung gelte für m. Betrachte paarweise disjunkte endliche Mengen $A_0, ..., A_m$. Dann sind $A_0 \cup ... \cup A_{m-1}$ und A_m disjunkt:

$$(A_0 \cup ... \cup A_{m-1}) \cap A_m = \emptyset.$$

Nach a) und der Induktionsannahme gilt dann

$$\begin{aligned} |A_0 \cup ... \cup A_m| &= |(A_0 \cup ... \cup A_{m-1}) \cup A_m| \\ &= |(A_0 \cup ... \cup A_{m-1})| + |A_m| \\ &= (|A_0| + ... + |A_{m-1}|) + |A_m| \\ &= |A_0| + ... + |A_m|. \end{aligned}$$

c) Die Mengen $A \setminus B$, $A \cap B$ und $B \setminus A$ sind paarweise disjunkt und es gelten die Gleichungen:

$$\begin{aligned} A &= (A \setminus B) \cup (A \cap B) \\ B &= (B \setminus A) \cup (A \cap B) \\ A \cup B &= (A \setminus B) \cup (A \cap B) \cup (B \setminus A) \end{aligned}$$

Nach a) und b) folgt daraus:

$$\begin{aligned} |A| &= |A \setminus B| + |A \cap B| \\ |B| &= |B \setminus A| + |A \cap B| \\ |A \cup B| &= |A \setminus B| + |A \cap B| + |B \setminus A| \\ &= (|A| - |A \cap B|) + |A \cap B| + (|B| - |A \cap B|) \\ &= |A| + |B| - |A \cap B|. \end{aligned}$$

d) Seien $A = \{a_0, ..., a_{m-1}\}$ und $B = \{b_0, ..., b_{n-1}\}$ mit jeweils paarweise verschiedenen Elementen $a_0, ..., a_{m-1}$ und $b_0, ..., b_{n-1}$. Dann ist

$$\begin{aligned} A \times B &= \{(a_i, b_j) | i = 0, ..., m-1 \text{ und } j = 0, ..., n-1\} \\ &= \bigcup_{i=0,...,m-1} \{(a_i, b_j) | j = 0, ..., n-1\} \end{aligned}$$

Die Mengen auf der rechten Seite sind paarweise disjunkt und haben alle die Kardinalität n. Nach b) gilt dann:

$$\begin{aligned} |A \times B| &= |\bigcup_{i=0,...,m-1} \{(a_i, b_j) | j = 0, ..., n-1\}| \\ &= \sum_{i=0,...,m-1} |\{(a_i, b_j) | j = 0, ..., n-1\}| \\ &= \sum_{i=0,...,m-1} n \\ &= m \cdot n \end{aligned}$$

e) lässt sich aus d) durch vollständige Induktion ähnlich b) zeigen. □

7.1 Das Dirichletsche Schubfachprinzip

Wenn $n+1$ Gegenstände in n Schubfächer gelegt werden, so gibt es in mindestens einem der Schubfächer mindestens 2 Gegenstände:

Satz 7.3. *Seien A und B endliche Mengen mit $|A| > |B|$ und $f\colon A \to B$ eine Funktion. Dann gibt es $a, a' \in A$, $a \neq a'$ mit $f(a) = f(a')$. Das bedeutet, dass die Abbildung $f\colon A \to B$ nicht injektiv ist.*

Beispiel 7.4. Unter 13 Personen haben mindestens 2 im gleichen Monat Geburtstag.

Das Prinzip lässt sich mit Berechnungen von Kardinalitäten endlicher Mengen kombinieren.

Beispiel 7.5. Wieviele verschiedene Familiennamen müssen in einem Telefonbuch vorkommen, damit es auf jeden Fall zwei Anschlüsse gibt, deren Namen dieselben ersten und zweiten Buchstaben haben? Wir definieren eine Abbildung f von den Anschlüssen in das Paar bestehend aus dem ersten und zweiten Buchstaben des zugehörigen Namens:

$$a_0 a_1 ... a_{m-1} \mapsto (a_0, a_1).$$

Die Bilder der Abbildung liegen in der Menge $\mathcal{A} \times \mathcal{A}$ aller Buchstabenpaare, wobei $\mathcal{A} = \{a, ..., z\}$ das Standardalphabet mit 26 Buchstaben ist. Nach vorangehenden Sätzen über Kardinalitäten ist $|\mathcal{A} \times \mathcal{A}| = 26 \cdot 26 = 676$. Wenn das Telefonbuch mindestens $676 + 1 = 677$ Einträge enthält, so gibt es zwei Anschlüsse, deren Namen dieselben ersten und zweiten Buchstaben haben.

Beispiel 7.6. Aus den Zahlen 1, 2, ..., 8 werden 5 Zahlen ausgewählt. Dann ist die Summe von zweien der ausgewählten Zahlen gleich 9. Sei A die Menge der 5 ausgewählten Zahlen, B die Menge

$$\{\{1,8\},\{2,7\},\{3,6\},\{4,5\}\}$$

der Zweiermengen, deren Summe 9 ist. Da $|B| = 4$ ist, gibt es zwei verschiedene Elemente in A, die Elemente desselben Elementes von B sind. Die Summe dieser zwei verschiedenen Elemente ist dann 9.

7.2 Zählformeln

Oft müssen Folgen von Elementen aus gegebenen Mengen betrachtet werden. Man muss unterscheiden, ob in diesen Folgen Wiederholungen erlaubt sind und dasselbe Element mehrfach vorkommen darf, und ob die Reihenfolge der Elemente berücksichtigt wird.

Beispiel 7.7. Aus der Menge $A = \{a, b, c\}$ sollen 2 Elemente ausgewählt werden:

- **Stichproben**: Wiederholungen erlaubt, Reihenfolge relevant: Dann gibt es $9 = 3 \cdot 3$ *Stichproben*:

$$(a,a),(a,b),(a,c),(b,a),(b,b),(b,c),(c,a),(c,b),(c,c).$$

- **Auswahlen**: Wiederholungen erlaubt, Reihenfolge irrelevant: Dann gibt es 6 *Auswahlen*:

$$(a,a),(a,b),(a,c),(b,b),(b,c),(c,c).$$

- **Permutationen**: Wiederholungen nicht erlaubt, Reihenfolge relevant: Dann gibt es 6 *Permutationen*:

$$(a,b),(a,c),(b,a),(b,c),(c,a),(c,b).$$

- **Kombinationen**: Wiederholungen nicht erlaubt, Reihenfolge irrelevant: Dann gibt es 3 *Kombinationen*:

$$(a,b),(a,c),(b,c).$$

Definition 7.8. *Sei $A=\{a_0,...,a_{n-1}\}$ eine endliche Menge und $k\in\mathbb{N}$ eine natürliche Zahl.*

*a) s ist eine k-**Stichprobe** aus A, wenn s eine Folge der Länge k mit Einträgen in A ist: $s\in A^k$.*

*b) s ist eine k-**Auswahl** aus A, wenn s eine Funktion $s\colon A\to\mathbb{N}$ ist mit*

$$s(a_0)+...+s(a_{n-1})=k$$

*c) s ist eine k-**Permutation** aus A, wenn $s=(s_0,\ ...,\ s_{k-1})$ eine injektive Folge der Länge k mit Einträgen in A ist, d.h. $\forall i<j<k\, s_i\neq s_j$.*

*d) s ist eine k-**Kombination** aus A, wenn s eine Teilmenge von A von der Kardinalität k ist.*

Derartige Auswahlen kommen in der Wahrscheinlichkeitstheorie vor. Ein Ereignis besteht in einer Auswahl von Elementen aus einer Menge, zur Bestimmung von Wahrscheinlichkeiten ist es erforderlich, die Anzahl der möglichen Auswahlen zu bestimmen.

Satz 7.9. *Sei $A=\{a_0,...,a_{n-1}\}$ eine endliche Menge der Kardinalität $n\geqslant 1$ und $k\in\mathbb{N}$ eine natürliche Zahl. Dann gilt*

a) $|\{s\,|\,s$ ist eine k-Stichprobe aus $A\}|=n^k$.

b) $|\{s\,|\,s$ ist eine k-Auswahl aus $A\}|=\frac{(n+k-1)!}{k!\,(n-1)!}$.

c) $|\{s\,|\,s$ ist eine k-Permutation aus $A\}|=\frac{n!}{(n-k)!}$.

d) $|\{s\,|\,s$ ist eine k-Kombination aus $A\}|=\frac{n!}{k!\,(n-k)!}$.

Beweis. a) Durch Induktion über $k\in\mathbb{N}$.
Induktionsanfang: $k=0$. Es gibt genau eine 0-Stichprobe aus A, nämlich die eindeutig bestimmte Folge der Länge 0. Ist die Anzahl der 0-Stichproben $=1=n^0$, und die Behauptung gilt für $k=0$.
Induktionsschritt: Betrachte $k\in\mathbb{N}$, und die Behauptung gelte für k. Dann ist

$$\{s\,|\,s \text{ ist } k+1\text{-Stichp. aus } A\} \;=\; \bigcup_{a\in A}\{s\,|\,s \text{ ist } k+1\text{-Stichpr. aus } A, s(k)=a\}.$$

Die rechte Seite ist eine disjunkte Vereinigung von Mengen. Die Menge

$$\{s \,|\, s \text{ ist } k+1\text{-Stichpr. aus } A, s(k) = a\}$$

ist genauso groß wie die Menge

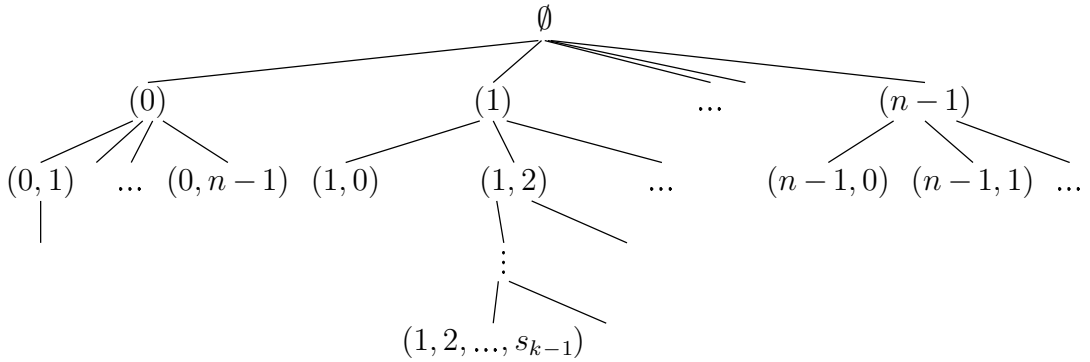
$$\{s \,|\, s \text{ ist } k\text{-Stichpr. aus } A\}.$$

Nach der Induktionsvoraussetzung hat diese Menge die Kardinalität n^k. Zusammen gilt:

$$\begin{aligned} |\{s \,|\, s \text{ ist } k+1\text{-Stichp. aus } A\}| &= \sum_{a \in A} |\{s \,|\, s \text{ ist } k\text{-Stichpr. aus } A\}| \\ &= \sum_{a \in A} n^k = n \cdot n^k = n^{k+1}. \end{aligned}$$

Also gilt die Behauptung für $k+1$.

Auch die anderen Behauptungen des Satzes lassen sich durch Induktion zeigen, wir wollen aber anschauliche Argumente benutzen.
c) Wir visualisieren die Menge der k-Permutationen $(s_0, \ldots, s_{k-1})$ aus A; zur Vereinfachung nehmen wir an, dass $A = \{0, 1, \ldots, n-1\}$:



Dieser Baum verzweigt bei der *Wurzel* $\emptyset$ n-fach, auf der nächsten Ebene $n-1$-fach, und allgemein auf der l-ten Ebene $n-l$-fach. Die k-Permutation stehen auf der k-ten Ebene. Die Anzahl der der Punkte auf der k-ten Ebene ergibt sich durch die Verzweigungsmöglichkeiten auf den Ebenen zuvor. Wir multiplizieren die Verzweigungsgrade auf den Ebenen $0, 1, \ldots, (k-1)$:

$$\begin{aligned} n \cdot (n-1) \cdots (n-(k-1)) &= \frac{n \cdot (n-1) \cdots (n-(k-1)) \cdot (n-k) \cdot (n-k-1) \cdots 1}{(n-k) \cdot (n-k-1) \cdots 1} \\ &= \frac{n!}{(n-k)!}. \end{aligned}$$

d) Sei X die Anzahl der k-Kombinationen aus A, d.h. die Anzahl der Teilmengen von A mit k Elementen. Diese Zahl steht in Beziehung zur gerade bestimmten Anzahl der k-Permutationen: Eine k-Permutation kann als eine k-Permutation einer k-elementigen Teilmenge von A aufgefasst werden:

$$\{s \,|\, s \text{ ist } k\text{-Permutation von } A\} = \bigcup_{\substack{B \subseteq A \\ |B| = k}} \{s \,|\, s \text{ ist } k\text{-Permutation von } B\}.$$

Die rechte Seite ist eine disjunkte Vereinigung, die Anzahl der Summanden ist die gesuchte Anzahl der k-Kombinationen aus A. Es gilt:

$$\begin{aligned}
|\{s \,|\, s \text{ ist } k\text{-Permutation von } A\}| &= \sum_{\substack{B\subseteq A\\ |B|=k}} |\{s \,|\, s \text{ ist } k\text{-Permutation von } B\}| \\
\frac{n!}{(n-k)!} &= \sum_{\substack{B\subseteq A\\ |B|=k}} \frac{k!}{(k-k)!} \\
&= X \cdot \frac{k!}{(k-k)!} = X \cdot k! \\
X &= \frac{n!}{k! \cdot (n-k)!}
\end{aligned}$$

b) Eine k-Auswahl aus A ist eine Funktion $s\colon A \to \mathbb{N}$ ist mit

$$s(a_0) + \ldots + s(a_{n-1}) = k\,.$$

Wir können s darstellen als eine geordnete Folge

$$(\underbrace{a_0, \ldots, a_0}_{s(a_0)}, \underbrace{a_1, \ldots, a_1}_{s(a_1)}, \ldots, \underbrace{a_{n-1}, \ldots, a_{n-1}}_{s(a_{n-1})}).$$

Diese Folge kann auch dargestellt werden durch Angabe der „Trenner" zwischen den Blöcken mit den Elementen von A, die wir als $|$ schreiben können:

$$(\underbrace{a_0, \ldots, a_0}_{s(a_0)}, |, \underbrace{a_1, \ldots, a_1}_{s(a_1)}, |, \ldots, |, \underbrace{a_{n-1}, \ldots, a_{n-1}}_{s(a_{n-1})}).$$

Die Folge hat die Länge $k + (n-1)$ ist durch die Positionen der Trenner eindeutig bestimmt:

$$(\ldots., |, \ldots, |, \ldots).$$

Diese Positionsfolge durch die Menge der Trennerpositionen angegeben werden. Diese Menge ist eine $n-1$-elementige Teilmenge einer Menge mit $k + (n-1)$ Elementen. Daher hat die Menge der k-Auswahlen aus A genausoviele Elemente wie die Menge der $n-1$-Kombinationen aus einer Menge mit $k + (n-1)$ Elementen. Diese Anzahl beträgt nach d):

$$\frac{(k+(n-1))!}{(n-1)! \cdot (k+(n-1)-(n-1)!} = \frac{(n+k-1)!}{k!(n-1)!}\,. \qquad \square$$

Beispiel 7.10. Das Geburtstagsparadoxon: Betrachte eine Menge von 20 Personen. Die Folge der Geburtstage (Tag und Monat) kann als 20-Stichprobe aus {1,..,365} aufgefasst werden. Es gibt 365^{20} solcher Folgen.

Die Menge der 20-Permutationen aus $\{1, \ldots, 365\}$ besteht aus allen Konfigurationen, in denen keine zwei Personen den gleichen Geburtstag haben. Es gibt

$$\frac{365!}{345!}$$

solcher Konfigurationen. Das Verhältnis von Konfigurationen mit paarweise verschiedenen Geburtstagen zu den Gesamtmöglichkeiten ist

$$\frac{365!}{345!}/365^{20} = \frac{365!}{345! \cdot 365^{20}}\,.$$

Wir bestimmen diese Zahl numerisch:

```
maxima] Float((365!)/((345!)*(365^20)))

(D18) 0.58856159741595


(C19)
```

Also gibt es bei mehr als 40% der Konfigurationen gemeinsame Geburtstage. Man kann diese Zahl als *Wahrscheinlichkeit* für das Auftreten gemeinsamer Geburtstage auffassen. Bei 50 Personen kanna man fast sicher sein, gemeinsame Geburtstage vorzufinden:

```
(C19) Float((365!)/((315!)*(365^50)))

(D24) 0.02962642037145


(C25)
```

Kapitel 8
Graphen

Definition 8.1. *Ein* ***schlichter Graph*** *ist eine Struktur $G = (E, K)$, wobei K eine zweistellige Relation auf E ist, die irreflexiv und symmetrisch ist:*

$$\forall x \in E \, \neg x K x \text{ und } \forall x, y \in E \, (x K y \rightarrow y K x).$$

Die Elemente der Trägermenge E heißen ***Ecken*** *von G. Die Paare $(a, b) \in K$ heißen* ***Kanten*** *von G; wenn $e = (a, b)$ eine Kante von G ist, so sind a und b in G* ***benachbart*** *oder* ***adjazent****, und die Kante e ist* ***inzident*** *zu a (und zu b). Der* ***Grad*** *einer Ecke a ist*

$$\delta(a) = |\{b \in E \mid a K b\}|.$$

Definition 8.2. *Sei $G = (E, K)$ ein schlichter Graph mit endlicher Trägermenge $E = \{e_0, ..., e_{m-1}\}$. Die* ***Adjazenzmatrix*** *von G ist eine Matrix $A = (a_{ij})_{0 \leqslant i,j < m}$ mit Einträgen $a_{ij} \in \{\boldsymbol{W}, \boldsymbol{F}\}$ aus der Menge der Wahrheitswerte und*

$$a_{ij} = \boldsymbol{W} \quad gdw. \quad e_i K e_j .$$

Die Adjazenzmatrix von G ist symmetrisch und alle Diagonaleinträge sind $= \boldsymbol{F}$.

Definition 8.3. *Sei $G = (E, K)$ ein schlichter Graph.*

a) *Eine Folge $(a_0, ..., a_{l-1})$ von Ecken ist ein* ***Weg*** *der* ***Länge*** *l in G, wenn*

$$\forall i < l - 1 \; a_i K a_{i+1} .$$

b) *Ein Weg $(a_0, ..., a_{l-1})$ ist ein* ***Zyklus*** *oder* ***Kreis*** *in G, wenn $l \geqslant 3$, $a_0, ...,$ a_{l-2} paarweise verschieden sind und $a_0 = a_{l-1}$.*

c) *Der Graph G is* ***azyklisch****, wenn er keinen Zyklus enthält.*

d) *Der Graph G ist* ***zusammenhängend****, wenn es für alle $a, b \in E$ einen Weg $(a, a_1, ..., a_{l-2}, b)$ in G gibt, wir sagen dass $(a, a_1, ..., a_{l-2}, b)$ ein Weg von a nach b ist.*

Betrachte eine Graphen $G = (E, K)$. Definiere eine Relation $\sim$ auf E, $a \sim b$ gdw. es einen Weg von a nach b gibt. Die Relation $\sim$ ist eine *Äquivalenzrelation* auf E:

a) Betrachte $a \in E$. (a) ist ein (trivialer) Weg von a nach a. Also ist $a \sim a$. Damit ist $\sim$ reflexiv.

b) Betrachte $a, b \in E$ mit $a \sim b$. Wähle einen Weg $(a, a_1, ..., a_{l-2}, b)$ von a nach b. Dann ist $(b, a_{l-2}, ..., a_1, a)$ ein Weg von b nach a. Also ist $b \sim a$. Damit ist $\sim$ symmetrisch.

c) Betrachte $a, b, c \in K$ mit $a \sim b$ und $b \sim c$. Wähle Wege $(a, a_1, ..., a_{l-2}, b)$ von a nach b und $(b, b_1, ..., b_{k-2}, c)$ von b nach c. Verkette diese Wege zum Weg

$$(a, a_1, ..., a_{l-2}, b, b_1, ..., b_{k-2}, c)$$

von a nach c. Also ist $a \sim c$. Damit ist $\sim$ transitiv.

Definition 8.4. *Sei $G = (E, K)$ ein Graph und definiere die Zusammenhangsrelation $\sim$ wie eben. Die Äquivalenzklassen von E nach $\sim$ sind die **Zusammenhangskomponenten** von G. Die Anzahl der Zusammenhangskomponenten von G ist die **Konnektivitätszahl** $c(G)$ von G.*

Beispiel 8.5. Beispiel eines Graphen und seiner Zusammenhangskomponenten.

Definition 8.6. *Sei $G = (E, K)$ ein Graph.*

a) *G ist ein **Eulerscher Graph**, wenn es einen Weg $(a_0, a_1, ..., a_{l-2}, a_{l-1})$ in G gibt, so dass $a_0 = a_{l-1}$ und in dem jede Kante (e, f) von G genau einmal vorkommt.*

b) *G ist ein **Hamiltonscher Graph**, wenn einen Zyklus $(a_0, a_1, ..., a_{l-1})$ in G gibt, in dem jede Ecke von G genau einmal vorkommt:*

$$a_0 = a_{l-1}\,, \forall a \in E \exists i < l\; a = a_i \text{ und } \forall i, j < l-1\,(i \neq j \to a_i \neq a_j).$$

8.1 Eulersche Graphen

Satz 8.7. *Sei $G = (E, K)$ ein zusammenhängender endlicher Graph mit $|E| \geqslant 2$. Dann ist G Eulersch genau dann, wenn jede Ecke $a \in E$ geraden Grad $\delta(a)$ hat.*

Beweis. Sei G Eulersch. Betrachte eine Ecke $a \in E$. Wähle einen Weg $(a_0, a_1, ..., a_{l-2}, a_{l-1})$ in G, so dass $a_0 = a_{l-1}$ und in dem jede Kante (e, f) von G genau einmal vorkommt. Wir können weiter annehmen, dass $a = a_0$. Die Menge der zu a inzidenten Kanten in G ist dann:

$$\{a\,a_1\,, a_{l-2}\,a\} \cup \bigcup_{1<i<l-2, a_i=a} \{a_{i-1}\,a_i\,, a_i\,a_{i+1}\}.$$

Da in dem Weg jede Kante genau einmal auftritt, ist dies eine disjunkte Vereinigung von jeweils zwei-elementigen Mengen. Ihre Kardinalität ist der Grad von a, diese ist eine gerade Zahl $\geqslant 2$.

Für die Umkehrung zeigen wir zunächst einen Hilfssatz:
(1) Sei $G' = (E', K')$ ein endlicher Graph, in dem jede Ecke e einen geraden Grad $\delta_{G'}(e)$ hat. Sei $a \in E'$ mit $\delta_{G'}(a) \geqslant 2$. Dann gibt es einen *geschlossenen* Weg $(a_0, a_1, ..., a_{l-2}, a_{l-1})$ in G' mit $l \geqslant 4$ und $a_0 = a_{l-1} = a$, in dem jede Kante $e\,f$ von G' höchstens einmal auftritt.
Beweis: Wähle einen Weg $w = (a_0, a_1, ..., a_{l-2}, a_{l-1})$ von maximaler Länge, so dass $a_0 = a$ und so dass jede Kante $e\,f$ von K' höchstens einmal in W auftritt. Angenommen, $a_{l-1} \neq a$. Dann kommt a_{l-1} unter den Kanten von w ungerade oft vor. Da der Grad von a_{l-1} gerade ist, gibt es eine Kante $a_{l-1}\,a_l$, die nicht in w vorkommt. Dann tritt in dem Weg $(a_0, a_1, ..., a_{l-2}, a_{l-1}, a_l)$ jede Kante höchstens einmal auf, im Widerspruch zur Maximalität von w. *qed*(1)

Wähle ein $a \in E$ und einen *geschlossenen* Weg $w = (a, a_1, ..., a_{l-2}, a)$ maximaler Länge, in dem jede Kante $e\,f$ von G höchstens einmal auftritt. Sei K_0 die Menge der in w vorkommenden Kanten von G. Angenommen $K_0 \neq K$. Dann setze

$$G' = (E, K \setminus K_0).$$

Wähle eine Kante $e\,f \in K \setminus K_0$. Da G zusammenhängend ist, wähle einen Weg p in G von a nach e.
Fall 1: Der Weg p verläuft vollständig in w. Dann ist der Endpunkt $e \in w$. Da $e\,f \in K \setminus K_0$, ist $\delta_{G'}(e) > 0$.
Fall 2: Es gibt eine Kante in p, die nicht in w liegt. Sei $b\,c \in K \setminus K_0$ die erste solche Kante entlang des Weges. Dann ist $b \in w$ und ähnlich wie oben ist $\delta_{G'}(b) > 0$.

In jedem Fall können wir also $b \in w$ wählen mit $\delta_{G'}(b) > 0$ hat. Anwendung von (1) auf G' liefert einen geschlossenen Weg $w' = (b, b_1, ..., b_{k-2}, b)$ in G', in dem jede Kante von G' höchstens einmal vorkommt. Wenn $b = a_i$ so ist

$$(a_0, ..., a_{i-1}, a_i = b, b_1, ..., b_{k-2}, b = a_i, a_{i+1}, ..., a_{l-2}, a_{l-1})$$

ein längerer Weg in G, in dem jede Ecke höchstens einmal vorkommt. Dies ist ein Widerspruch zur Maximalität von w. □

Aus dem vorangehenden Beweis kann man einen *Algorithmus* zur Konstruktion eines Eulerschen Wegs in einem endlichen zusammenhängenden G extrahieren: man finde zunächst einen geschlossenen Weg, in dem jede Kante höchstens einmal vorkommt. Dann suche man einen Punkt auf dem Weg, von dem eine Kante abgeht, die nicht in dem Weg liegt. Wenn es keinen solchen Punkt gibt, ist man fertig. Ansonsten bilde einen Weg in G' wie im Beweis und verkette diesen mit dem ursprünglichen Weg. Durch Iteration dieses Verfahrens wird der Graph ausgeschöpft und man erhält einen Eulerschen Pfad.

Die Frage nach Eulerschen Wegen in Graphen ist also recht gut algorithmisch beherrschbar. Ein vorgelegter Graph G wird auf Zusammenhang überprüft: ausgehend von einer beliebig gewählten Ecke a wird sukzessiv die Zusammenhangskomponente von a erzeugt. Wenn diese den gesamten Graphen ausschöpft, ist G zusammenhängend. Der Graph G ist weiterhin Eulersch, wenn jede Ecke geraden Grad besitzt.

8.2 Hamiltonsche Graphen

Hamiltonsche Graphen lassen sich algorithmisch nicht einfach beherrschen, es gibt noch viele ungelöste Probleme in diesem Zusammenhang. Die Entscheidung, ob ein vorgelegter Graph Hamiltonsch ist, kann bisher nicht durch einen allgemeinen Algorithmus getroffen werden. Wir verdeutlichen das am Beispiel eines Graphen, bei dem die Eigenschaft nicht-Hamiltonsch zu sein durch ein etwas verwickeltes Argument gezeigt werden kann:

Definition 8.8. *Ein **schlichter gewichteter Graph** ist eine Struktur $G = (E, \mathbb{R}, K, w)$, wobei (E, K) ein schlichter Graph ist und $w\colon E \times E \to \mathbb{R}$, so dass*

$$\forall x, y \in E\, w(x, y) = w(y, x).$$

*Wenn $xy \in K$ so ist $w(x, y) = w(xy)$ das **Gewicht** der Kante xy. Wenn $(a_0, ..., a_{l-1})$ ein Weg in G ist, so ist das **Gewicht** von G definiert als*

$$w(a_0, ..., a_{l-1}) = \sum_{i=0}^{l-2} w(a_i, a_{i+1}).$$

Gewichte können als Aufwand für das Zurücklegen einer Kante betrachtet werden, etwa als Zeit oder Entfernung. Das *Problem des Handlungsreisenden* ist folgende Aufgabe:

> Zu einem Hamiltonschen Graphen G finde man einen Hamiltonschen Pfad mit minimalem Gewicht.

Das Problem ist in dieser Form allgemein nicht mit vertretbarem Rechenaufwand lösbar. In seiner allgemeinen Form ist die Lösung des Problems in polynomieller Zeit äquivalent zu $P = \mathrm{NP}$. Wegen der grundsätzlichen Bedeutung des Problems, nicht nur für Handlungsreisende, sind eine Reihe von effizienten Näherungsalgorithmen entwickelt worden, die allgemein oder auf Spezialfälle zutreffen. Ein primitiver *suboptimaler* Algorithmus ist es, an jeder Stelle zum nächsten Nachbarn weiterzulaufen, d.h. die Kante von geringstem Gewicht zu benutzen. Dieser Algorithmus wird aber im allgemeinen keinen Hamiltonschen Weg finden, oder einen Hamiltonschen Weg mit suboptimalem Gewicht.

8.3 Bäume

Definition 8.9. *Ein Graph $G = (E, K)$ ist ein **Baum**, wenn G zusammenhängend und azyklisch ist.*

Satz 8.10. *Sei $G = (E, K)$ ein endlicher Baum mit n Ecken und m Kanten. Dann sind folgende Aussagen äquivalent:*

a) G ist ein Baum.

*b) Zwischen zwei beliebigen Ecken von G existiert **genau ein** Weg.*

c) G ist zusammenhängend, und wenn man eine beliebige Kante aus G entfernt, so ist G nicht mehr zusammenhängend.

d) G ist zusammenhängend und $m = n - 1$.

e) G ist azyklisch, und durch Hinzufügen einer neuen Kante erhält G einen Zyklus.

Beweis. *a)* $\rightarrow$ *b)* Sei G Baum. Angenommen, es gibt Ecken a und b und zwei verschiedene Wege $w = (a, a_1, ..., a_{l-2}, b)$ und $w' = (a, a'_1, ..., a'_{l'-2}, b)$ von a nach b. Da $w \neq w'$ können wir ein kleinstes $i+1$ wählen mit $a_{i+1} \neq a'_{i+1}$. Wähle dann minimale Indizes $j, k > i$ mit $a_j = a'_k$. Dann ist

$$(a_i, ..., a_j = a'_k, a'_{k-1}, ..., a'_i = a_i)$$

ein Zyklus in G, im Widerspruch zur Azyklizität von G.

b) $\rightarrow$ *c)* Es gelte b). Betrachte eine Kante ab in G. Nach b) ist dies der einzige Weg von a nach b. Nach Entfernen von ab gibt es keinen Weg mehr von a nach b und der Graph wird unzusammenhängend.

c) $\rightarrow$ *d)* Wir beweisen die Implikation durch Induktion über die Mächtigkeit $|E|$ von G. Betrachte einen Graphen $G = (E, K)$, der c) erfüllt, und die Implikation gelte für alle kleineren Graphen als G. Wähle eine Kante $ab \in E$.
(1) Der Graph $G' = (E, K \setminus \{ab\})$ hat genau zwei Zusammenhangskomponenten: die durch a bzw. b bestimmt werden.
Beweis: Angenommen, die Ecken a und b lassen sich in G' durch einen Weg p verbinden. Betrachte beliebige Ecken e, $f \in E$. Da G zusammenhängend ist, gibt es in G einen Weg w von e nach f. Falls die Kante ab in w vorkommt, ersetze diese jeweils durch den Weg p. Der substituierte Weg ist ein Weg von e nach f in G'. Also ist G' zusammenhängend, im Widerspruch zu c).

Betrachte nun eine beliebige Ecke $e \in E$. Da G zusammenhängend ist, gibt es in G einen Weg w von e nach a.
Fall 1: w enthält die Kante ab nicht. Dann ist w ein Weg in G' von e nach a. Also liegt e in der Zusammenhangskomponente von a in G'.
Fall 2: w enthält die Kante ab. Sei $w' = (e, ..., f)$ maximales Anfangsstück von w, das die Kante ab nicht enthält. Dann ist $f = a$ oder $f = b$, w' ist ein Weg in G', und e ist in der Zusammenhangskomponente von a oder von b in G'. *qed*(1)

Sei G_a die Zusammenhangskomponente von a und G_b die Zusammenhangskomponente von b bezüglich G'.
(2) G_a und G_b erfüllen c).
Beweis: Als Zusammenhangskomponenten sind G_a und G_b zusammenhängend. Betrachte eine beliebige Kante ef in G_a . Die Ecken e und f sind dann nicht mehr in $K \setminus \{ef\}$ durch einen Weg verbindbar. Dies impliziert, dass e und f nach Fortlassen von ef auch nicht in G_a verbindbar sind. *qed*(2)

Nach Induktionsvoraussetzung gilt *d)* für G_a und G_b : Wenn n_a bzw. m_a die Anzahl der Ecken und Kanten von G_a ist, so gilt $m_a = n_a - 1$. Entsprechend ist $m_b = n_b - 1$. Da die Trägermenge des Graphen G die disjunkte Vereinigung der Trägermengen der Graphen G_a bzw. G_b ist, ist

$$n = n_a + n_b .$$

Wir hatten schon oben überlegt, dass die einzige gewöhnliche Verbindung von G_a und G_b ist entlang der Kante ab. Es gibt keine weiteren Kanten, die G_a udn G_b verbinden. Für die Anzahlen der Kanten gilt:

$$m = m_a + m_b + 1.$$

Nach Induktionsvoraussetzung ist $m_a = n_a - 1$ und $m_b = n_b - 1$. Damit ist

$$m = m_a + m_b + 1 = n_a - 1 + n_b - 1 + 1 = (n_a + n_b) - 1 = n - 1.$$

Also gilt d) auch für G.

d) $\rightarrow$ *e)* Wir beweisen die Implikation durch Induktion über die Eckenzahl $n \in \mathbb{N} \setminus \{0\}$.
Induktionsanfang: Sei $n = 1$. Dann ist G der triviale Graph $\bullet$ mit einer Ecke. G erfüllt die Eigenschaft e) aus trivialen Gründen.
Induktionsschritt: Angenommen, die Implikation gilt für n. Betrachte einen zusammenhängenden Graphen $G = (E, K)$ mit $n+1$ Ecken und n Kanten.
(3) Es gibt eine Ecke $a \in E$ mit $\delta_G(a) = 1$.
Beweis. Da G zusammenhängend ist und $|E| \geqslant 2$, ist $\forall a \in E\, \delta_G(a) \geqslant 1$. Angenommen es gilt $\forall a \in E\, \delta_G(a) \geqslant 2$. Zu jeder Ecken gehören dann zwei Kanten, von denen jede wiederum zu zwei Ecken adjazent ist. Damit ist

$$(n+1) \cdot \frac{2}{2} = n+1$$

eine untere Schranke für die Anzahl der Kanten, im Widerspruch zur Voraussetzung. *qed*(3)

Wähle ein $a \in E$ mit $\delta_G(a) = 1$. Wähle das eindeutig bestimmte $b \in E$ mit $ab \in K$. Wir entfernen die Ecke a und die Kante ab aus G. Der Graph

$$G' = (E \setminus \{a\}, K \setminus \{ab\})$$

ist zusammenhängend mit Eckenzahl n und Kantenzahl $n-1$. Nach Induktionsvoraussetzung erfüllt G' die Eigenschaft e): G' ist azyklisch und durch Hinzufügen einer neuen Kante erhält G' einen Zyklus.
(4) G ist azyklisch.
Beweis: Angenommen, $z = (a_0, ..., a_{l-1} = a_0)$ ist ein Zyklus in G. Jede Ecke in einem Zyklus hat mindestens Grad 2. Da $\delta_G(a) = 1$ ist, kommt a nicht in z vor. Also ist z ein Zyklus in G', im Widerspruch zur Azyklizität von G'. *qed*(4)
(5) Seien $e, f \in E$, $e \neq f$ und $ef \notin K$. Dann enthält der Graph $G^+ = (E, K \cup \{ef\})$ einen Zyklus.
Beweis: Wenn $e, f \in E \setminus \{a\}$, so ist ef eine neue Kante für G'. G' enthält nach Hinzufügen von ef einen Zyklus, der auch Zyklus in G^+ ist. Andernfalls können wir annehmen, dass $e = a$. Dann ist $f \neq b$ und $f, b \in E \setminus \{a\}$. Da G' zusammenhängend ist, gibt es einen Weg $(f = a_0, a_1, ..., a_{l-1} = b)$ von minimaler Länge, der f und b verbindet. Dann ist $(e = a, f = a_0, a_1, ..., a_{l-1} = b, a)$ ein Zyklus in dem Graphen G^+. *qed*(5)

e) $\rightarrow$ *a)* Angenommen, G erfüllt e). Für a) genügt es zu zeigen, dass G zusammenhängend ist. Betrachte Ecken $a, b \in E$. Falls $ab \in K$, so gibt es trivialerweise einen Weg von a nach b. Falls $ab \notin K$, füge ab als neue Kante zu G hinzu. Dann ist der erweiterte Graph zyklisch. Wähle einen Zyklus $z = (a_0, a_1, ..., a_{l-1})$ im erweiterten Graphen. Da der ursprüngliche Graph azyklisch war, muss die neue Kante ab in z vorkommen. Da z ein Zyklus ist, kommt ab genau einmal in z vor. Ohne Einschränkung ist $a = a_0 = a_{l-1}$ und $b = a_1$. Dann ist $(b = a_1, ..., a_{l-1} = a)$ ein Weg von b nach a im Graphen G. Also ist G zusammenhängend. □

Jeder Graph enthält Teilbäume, z.B. die einpunktigen Teilgraphen. Von besonderem Interesse sind maximale Teilbäume:

Definition 8.11. *Sei $T = (E, K')$ ein Teilgraph von $G = (E, K)$, d.h. $K' \subseteq K$. Dann ist T ein **Spannbaum** von G, wenn T ein Baum ist. Ein Spannbaum ist also ein Teilbaum, der alle Ecken von G umfasst.*

Zu einem gegebenen Graphen gibt es in der Regel viele Spannbäume. Ein Problem, das auf den ersten Blick dem Problem des Handlungsreisenden ähnelt, ist die Bestimmung eines bezüglich einer Wichtung minimalen Spannbaums:

Definition 8.12. *Sei $G = (E, \mathbb{R}, K, w)$ ein endlicher gewichteter Graph. Ein **minimaler Spannbaum** ist ein Spannbaum $T = (E, K')$ von G, dessen Gesamtgewicht*

$$\sum_{ab \in K'} w(a, b) \in \mathbb{R}$$

minimal ist.

,