

Modellierung

Prof. Dr. Sibylle Schwarz

HTWK Leipzig, Fakultät IM

Gustav-Freytag-Str. 42a, 04277 Leipzig

Zimmer Z 411 (Zuse-Bau)

<https://informatik.htwk-leipzig.de/schwarz>

sibylle.schwarz@htwk-leipzig.de

Wintersemester 2025/26

Informatik

Informatik Lehre von der **symbolischen Darstellung** und **Verarbeitung** von **Information** durch **Algorithmen**

Teilgebiete der Informatik:

theoretisch

- ▶ Sprachen zur Formulierung von Information und Algorithmen,
- ▶ Möglichkeiten und Grenzen der Berechenbarkeit durch Algorithmen,
- ▶ Grundlagen für technische und praktische (und angewandte) Informatik

technisch

- ▶ maschinelle Darstellung von Information
- ▶ Mittel zur Ausführung von Algorithmen

(Rechnerarchitektur, Hardware-Entwurf, Netzwerk, ...)

praktisch Entwurf und Implementierung von Algorithmen
(Betriebssysteme, Compilerbau, SE, ...)

angewandt Anwendung von Algorithmen
(Text- und Bildverarbeitung, Datenbanken, KI, Medizin-, Bio-, Wirtschafts-, Medien-Informatik, ...)

Beispiel Münzenspiel

Spielfeld:

(unendlich viele) Stapel von Münzen nebeneinander



1. Zu Beginn liegen 5 Münzen auf einem Stapel, alle anderen Stapel sind leer.
2. In jedem Zug werden zwei Münzen von einem Stapel (auf dem wenigstens zwei Münzen liegen) genommen und eine davon auf den rechten, die andere auf den linken Nachbarstapel gelegt.
3. Das Spiel ist zuende, wenn kein Zug mehr möglich ist.

Münzenspiel: Fragen

- ▶ In welchen Spielzuständen sind keine Züge möglich?
- ▶ Welche Zustände sind aus dem Startzustand erreichbar?
- ▶ Wieviele Züge können (mindestens / höchstens) gespielt werden, bis kein Zug mehr möglich ist?

Münzenspiel für zwei Personen mit den zusätzlichen Regeln:

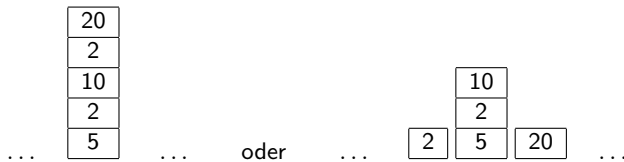
4. Beide Spieler ziehen abwechselnd.
 5. Wer am Zug ist, wenn kein Zug mehr möglich ist, verliert.
(Der andere gewinnt.)
- ▶ Kann der Spieler, der den ersten Zug macht, gewinnen?
 - ▶ Wie kann der Spieler, der den ersten Zug macht, gewinnen?
 - ▶ Gewinnt immer der Spieler, der den ersten Zug macht?
 - ▶ (Wie) Hängt das von der Anzahl der Münzen zu Beginn ab?

Modellierung des Münzenspiels: Zustände



Zustand (Konfiguration): Momentaufnahme des Spielfeldes
vor oder nach einem (vollständigen) Spielzug
formale (maschinenlesbare) Modellierung eines Spielzustandes, z.B. durch

- Folge von Stapeln (Folgen) von Münzen verschiedener Werte, z.B.



- Folge natürlicher Zahlen (Anzahlen der Münzen je Stapel),
z.B. [..., 0, 5, 0, ...] oder [..., 0, 1, 3, 1, 0, ...]
- endliche Folge natürlicher Zahlen (nur relevanter Bereich),
z.B. [5] oder [1, 3, 1] oder [0, 5, 0] oder [0, 1, 3, 1, 0]

Abstraktion von Art und Anzahl der Objekte
ermöglicht einfache Übertragung auf ähnliche Aufgaben

Modellierung des Münzenspiels: Ablauf

Modellierung aller möglichen Übergänge zwischen Zuständen
z.B. als

- ▶ formale Darstellung der Spielregeln als (beidseitig unendliche) Folge mit den Positionen $\mathbb{Z} = [\dots, -2, -1, 0, 1, 2, \dots]$

Startzustand : $[\dots, 0, 5, 0 \dots]$

Spielzug: Ersetzung einer Teilfolge des Zustandes der Form $[x, y + 2, z]$ durch $[x + 1, y, z + 1]$

Regel: $[x, y + 2, z] \rightarrow [x + 1, y, z + 1]$

Spielende , wenn jedes Element der Folge < 2
(kein Zug mehr möglich)

- ▶ formale (graphische) Darstellung aller möglichen Spielabläufe

Modellierung des Verlaufes eines Spieles z.B. als

- ▶ Weg im Spielgraphen
- ▶ (gültige) Folge von Positionen in $\mathbb{Z}$,
an der die (hier einzige) Regel angewendet wurde

Modelle

Modelle sind **Abstraktionen** (Vereinfachungen)
realer Dinge, Eigenschaften, Beziehungen, Vorgänge

- ▶ Auswahl der (für den Anwendungsbereich, zur Problemlösung) wichtigen Informationen
- ▶ Vernachlässigung unwichtiger Details

Beispiele:

- ▶ Spielzustände im Münzspiel:
 - wichtig: lineare Anordnung der Positionen (Spielfeld), Zuordnung von (Münz-)Anzahlen zu Positionen,
 - unwichtig: Art und Werte der Münzen, Abstände der Stapel, ...
- ▶ Liniennetzplan
 - wichtig: Stationen, Verbindungen zwischen Stationen, Art der Verbindung (z.B. Liniennummer)
 - unwichtig: genaue Linienführung, aktuelle Verspätungen, Baustellen, ...
- ▶ Grundriss, Stundenplan, Ablaufplan, Kostenplan
- ▶ Holzmodell eines Gebäudes, Modellfahrzeug

Art der Modelle abhängig von **Problemereich** und geplanter **Verwendung**

Prozess beim Lösen von Aufgaben (Problemen)

Analyse der (informalen) Aufgabe, Identifikation von

- ▶ Aufgabenbereich (Kontext)
- ▶ Frage (Typ, mögliche Werte, Eigenschaften der Eingabedaten)
- ▶ gewünschte Lösung (Ausgabe: Typ, Eigenschaften, Zusammenhang mit Eingabe)

Modellierung (Abstraktion und formale Darstellung) von

- ▶ Aufgabenbereich (Kontext)
- ▶ Anforderungen an Eingaben
- ▶ Anforderungen an Lösungen

Modellierung von **Daten**

und deren **Eigenschaften** und **Beziehungen** zueinander

Entwurf einer Lösungsstrategie für die modellierte Aufgabe
(mit vorhandenen oder neuen Methoden)

Modellierung von **Abläufen und deren Eigenschaften**

Umsetzung der Lösungsstrategie im Modellbereich

Ausführung der Lösungsstrategie im Modellbereich

Übertragung der Lösung vom Modellbereich in die Realität

Formalisierung: Syntax und Semantik

algorithmische (maschinelle) **Verarbeitung** von Information
erfordert geeignete **Darstellung** der Information

- ▶ Formalismus (Kalkül):
formale Sprache, Maschinen-(und Menschen-)lesbar
- ▶ eindeutige Zuordnung einer Bedeutung zu den Elementen des Formalismus

Semantik (Bedeutung)

Was wird dargestellt?

Syntax (Darstellungsform)

Wie wird es dargestellt?

(meist viele verschiedene Möglichkeiten)

Beispiel: Münzspiel

Formale Methoden

Abstraktion von unwichtigen Details
(Entwicklung und Verwendung von Modellen)

Präzisierung der relevanten Aussagen
(eindeutige Semantik)

Systematisches Lösen (auch maschinell) von formal dargestellten
Problemen möglich

Struktureigenschaften formaler Beschreibungen
Schlussweisen unabhängig von Bedeutung der
Aussagen

Aus der Modulbeschreibung

Modul C114 Modellierung (8 ECTS-Punkte)

Arbeitsaufwand: Präsenzzeit 6 SWS = 4 SWS V + 2 SWS S
Selbststudienzeit 156 h

Lernziele: Die Studierenden können mathematische und logische Grundkonzepte zur Modellierung praktischer Aufgabenstellungen anwenden.
Sie können Anforderungen an Software und Systeme formal beschreiben und wissen, dass deren Korrektheit mit formalen Methoden nachweisbar ist.

Inhalt der Lehrveranstaltung Modellierung

Einführung in formale Beschreibungsverfahren in der Informatik
Modellierung von

Aussagen durch aussagenlogische Formeln

Daten durch

- ▶ Mengen
- ▶ Wörter (Folgen) und Sprachen
- ▶ Terme

Zusammenhängen (Beziehungen) durch

- ▶ Relationen
- ▶ Graphen
- ▶ Strukturen
- ▶ Abstrakte Datentypen

Abläufen durch Zustandübergangssysteme

Eigenschaften, Anforderungen (für Daten und Systeme) durch
Formeln der Prädikatenlogik (der ersten Stufe)
(Ausblick in nichtklassische Logiken)

Lernziele (und Nebenwirkungen)

- ▶ Fähigkeit zur Abstraktion
- ▶ Verständnis der grundlegenden Modellierungs-Formalismen der Informatik:
Logik, Mengen, Relationen, Graphen, Terme, Strukturen, Datentypen
- ▶ anwendungsbereite Kenntnisse zur Modellbildung
- ▶ Zusammenhänge zu anderen Gebieten der Informatik und zur Mathematik

Literatur

Folien zur Vorlesung, jeweils nach der Vorlesung veröffentlicht

[https:](https://informatik.htwk-leipzig.de/schwarz/lehre/ws25/modellierung)

[//informatik.htwk-leipzig.de/schwarz/lehre/ws25/modellierung](https://informatik.htwk-leipzig.de/schwarz/lehre/ws25/modellierung)

empfohlene Bücher:

- ▶ zur Modellierung:
 - ▶ Uwe Kastens, Hans Kleine Büning: Modellierung - Grundlagen und formale Methoden, Hanser 2008
- ▶ zur Logik
 - ▶ Michael Huth, Mark Ryan: Logic in Computer Science, Cambridge University Press 2010
 - ▶ Uwe Schöning: Logik für Informatiker, Spektrum, 1995
 - ▶ Martin Kreuzer, Stefan Kühling: Logik für Informatiker, Pearson Studium, 2006

Organisation der Lehrveranstaltung

Folien, Übungsserien, Termine, Änderungen, ... unter <https://informatik.htwk-leipzig.de/schwarz/lehre/ws25/modellierung>

Präsenzstudium

6 SWS

Lehrveranstaltungen für jeden Studenten:

Vorlesung jeden Donnerstag und Freitag (4SWS)

Seminar (5 Gruppen) Montag bis Mittwoch (2SWS)

Gäste, Wiederholer (vor JG 25): mit **MIB**-Gruppen

- ▶ Besprechung der Übungsserien (Vorrechnen)
- ▶ Fragen zum aktuellen Vorlesungsinhalt

Selbststudium: (incl. PV) 156 h $\approx$ 10 h/Woche

Vor- und Nachbereitung der Vorlesungen anhand

Folien und angegebener Literatur

Übungsaufgaben zu jeder Vorlesung (wöchentlich einzusenden)

klar, überschaubar, Feedback

schriftliche Übungsserien (vor Donnerstag)

praktische Aufgaben im Autotool (vor Montag)

Literaturstudium ergänzend, Beitrag zur Lösung der Aufgaben

Selbststudium

Vor- und Nachbereitung **jeder** Lehrveranstaltung

(Vorlesung, Seminar, Praktikum, ...)

Unterlagen zu den Lehrveranstaltungen (Folien, eigene Notizen)

durcharbeiten

(enthalten auch zum Lösen der Übungsaufgaben notwendige Definitionen, Herleitungen, Beispiele, ...)

Übungsaufgaben **regelmäßig** und **rechtzeitig** lösen,

(Aufgaben vom Typ der) Übungsaufgaben gehören zum Inhalt des Moduls und werden geprüft

Fachliteratur benutzen (z.B. Bücher, E-Books in Bibliothek),
enthalten Erklärungen, zusätzliche Übungsaufgaben
(Internet-Quellen sind oft unzuverlässig)

Lerngruppen bilden und **gemeinsam** lernen und Aufgaben lösen

Nachfragen bei Dozenten (E-Mail, Sprechzeit, nach der
Lehrveranstaltung, ...), Mitstudenten, älteren
Studenten, Fachschaft, ...

Schriftliche Übungsaufgaben (Übungsserien)

- ▶ gestellt jeweils am Donnerstag
(Aufgaben zu den Vorlesungen am Donnerstag und Freitag)
- ▶ Lösungen bis zum folgenden Mittwoch ausschließlich online über OPAL einzusenden (wie in der Einführungswoche erklärt)
genau **eine pdf-Datei je Aufgabe**, Dateiname serieX-aufgY-Z.pdf
für Lösung zu Aufgabe Y von Übungsserie X vom den Studierenden mit Familiennamen Z,
- ▶ Lösung in Gruppen aus ≤ 3 Studierenden zulässig (und empfohlen),
Namen **aller Mitglieder der Gruppe** oben auf der Lösung vermerken,
jedes Gruppenmitglied sendet die (gemeinsame) Lösung zu Opal
Dateiname serieX-aufgY-Z1-Z2-Z3.pdf
- ▶ Bewertung bis zum folgenden Sonntag
(0: falsch, 1: überwiegend richtig, 2: korrekt)
- ▶ Besprechung in der darauffolgenden Übung
Vorträge (Vorrechnen) zu den Aufgabenlösungen

Übungsserien auch unter <https://informatik.htwk-leipzig.de/schwarz/lehre/ws25/modellierung>

Nicht im Opal-Kurs eingesendete Lösungen werden **nicht gewertet**.

Seminar (Übungen)

Lernziele bei der Bearbeitung der Übungsaufgaben:

- ▶ Nachbereitung der letzten Vorlesung anhand der Vorlesungsfolien und Literatur (z.B. der angegebenen)
- ▶ Vorbereitung der nächsten Vorlesung
- ▶ Vorbereitung der Seminarvorträge zu jeder Aufgabe

Seminar (Übungen):

- ▶ Besprechung der Lösungen der schriftlichen Übungsaufgaben, Vorrechnen zur Prüfungszulassung
- ▶ Fragen zum aktuellen Vorlesungsstoff und zu den neuen schriftlichen und praktischen Übungsaufgaben
- ▶ **in jede Übung mitbringen:** (alles sofort les- und auffindbar) alle bisherigen Vorlesungsfolien und eigene Notizen dazu, Übungsserien und eigene Lösungen

Praktische Hausaufgaben – Autotool

<https://autotool.imn.htwk-leipzig.de/new>

(Einführung am vergangenen Donnerstag)

einmal zu Beginn:

- ▶ Anmeldung über Shibboleth (HRZ-Login)
- ▶ Click „Einschreibung“,
eigene Übungsgruppe auswählen

jedes Mal:

- ▶ Anmeldung über Shibboleth (HRZ-Login)
- ▶ Aufgabe ansehen (Click „Bearbeiten“)
- ▶ Aufgabe lösen (meist mit Stift, Papier, Folien, Literatur),
Lösung in Textfeld eintragen,
Click „Textfeld absenden“
- ▶ Autotool-Antwort lesen und verstehen !

Prüfung

Zulassungsvoraussetzungen (Prüfungsvorleistungen):
regelmäßige erfolgreiche Lösung der Übungsaufgaben,
also **alle** folgenden drei Bedingungen:

- ▶ rechtzeitige Einsendung von richtigen Lösungen
(mit wenigstens 1 Punkt bewertet) zu wenigstens 70% aller
gestellten schriftlichen Übungsaufgaben (OPAL)
- ▶ Präsentation: mindestens drei Vorträge in den Seminaren
(richtiges „Vorrechnen“ der Lösungen)
je einmal bis Ende November, Dezember und Januar
- ▶ 50% aller Punkte für praktische Pflichtaufgaben (Autotool)

Prüfung: Klausur 120 min

Aufgabentypen aus den Übungsserien

einziges zugelassenes Hilfsmittel:

ein beidseitig handbeschriebenes A4-Blatt

Aussagenlogik

Aussagen

Aussage = Behauptung

Beispiele:

- ▶ Es regnet.
- ▶ Die Straße ist nass.
- ▶ 9 ist eine Primzahl.
- ▶ $\sqrt{2} \in \mathbb{Q}$
- ▶ $3 < 5$
- ▶ $x < 5$ (hängt von x ab, keine Aussage)
- ▶ Ist $x < 5$? (keine Aussage)
- ▶ Sei $x < 5$. (keine Aussage)
- ▶ Morgen regnet es.
- ▶ Es ist nicht alles Gold, was glänzt.

Wahrheitswerte

Prinzipien der **klassischen** Logik:

Zweiwertigkeit: Jede Aussage ist wahr oder falsch.

ausgeschlossener Widerspruch:

Keine Aussage ist sowohl wahr als auch falsch.

Wahrheitswerte 1 (wahr) oder 0 (falsch)

Jede Aussage p hat genau einen Wahrheitswert $W(p) \in \{0, 1\}$.

Beispiele:

- ▶ $W(\text{Es regnet.}) = ?$
- ▶ $W(\text{Die Straße ist nass.}) = ?$
- ▶ $W(9 \text{ ist eine Primzahl.}) = 0$
- ▶ $W(\sqrt{2} \in \mathbb{Q}) = 0$
- ▶ $W(3 < 5) = 1$
- ▶ $W(\text{Morgen regnet es.}) = ?$
- ▶ $W(\text{Es ist nicht alles Gold, was glänzt.}) = 1$

Zusammengesetzte Aussagen – Junktoren

Junktor (mit zugeordneter Stelligkeit):

Symbol (Syntax) für **Verknüpfung** von Aussagen

z.B. „und“ (zweistellig), „nicht“ (einstellig)

Gottlob Frege (1848–1925):

Die Bedeutung des Ganzen ist eine Funktion der Bedeutung seiner Teile.

Der Wahrheitswert einer zusammengesetzten Aussage lässt sich aus den Wahrheitswerten ihrer Teilaussagen berechnen.

Semantik (Bedeutung) eines n -stelligen Junktors $*$:

$\llbracket * \rrbracket : \{0, 1\}^n \longrightarrow \{0, 1\}$

(n -stellige Funktion auf der Menge $\{0, 1\}$)

Wahrheitswertkonstanten (nullstellige Junktoren):

$\mathfrak{t}$ mit $\llbracket \mathfrak{t} \rrbracket = 1$

$\mathfrak{f}$ mit $\llbracket \mathfrak{f} \rrbracket = 0$

Konjunktion $\wedge$

Es regnet **und** 9 ist eine Primzahl.

- ▶ $W(9 \text{ ist eine Primzahl.}) = 0$
- ▶ $W(\text{Es regnet.}) = ?$
- ▶ $W(\text{Es regnet und } 9 \text{ ist eine Primzahl.}) = 0$

$p \wedge q$ ist genau dann wahr,
wenn **beide Aussagen p und q** wahr sind.

$W(p)$	$W(q)$	$W(p \wedge q)$
0	0	0
0	1	0
1	0	0
1	1	1

$$W(p \wedge q) = \min(W(p), W(q))$$

$\llbracket \wedge \rrbracket = \min$ ist kommutativ, assoziativ

$$\bigwedge_{i=1}^n p_i = p_1 \wedge p_2 \wedge \cdots \wedge p_n$$

Disjunktion $\vee$ (inklusive)

Es regnet oder $3 < 5$.

- ▶ $W(3 < 5) = 1$
- ▶ $W(\text{Es regnet}) = ?$
- ▶ $W(\text{Es regnet oder } 3 < 5.) = 1$

$p \vee q$ ist genau dann wahr,
wenn **wenigstens eine der Aussagen p und q** wahr ist.

$W(p)$	$W(q)$	$W(p \vee q)$
0	0	0
0	1	1
1	0	1
1	1	1

$$W(p \vee q) = \max(W(p), W(q))$$

$\llbracket \vee \rrbracket = \max$ ist kommutativ, assoziativ

$$\bigvee_{i=1}^n p_i = p_1 \vee p_2 \vee \cdots \vee p_n$$

Negation $\neg$

$$\neg(\sqrt{2} \in \mathbb{Q}) \quad (\text{oft auch } \sqrt{2} \notin \mathbb{Q})$$

$$\blacktriangleright W(\sqrt{2} \in \mathbb{Q}) = 0$$

$$\blacktriangleright W(\neg(\sqrt{2} \in \mathbb{Q})) = 1$$

$\neg p$ ist genau dann wahr, wenn p falsch ist.

$W(p)$	$W(\neg p)$
0	1
1	0

$$W(\neg p) = 1 - W(p)$$

Implikation $\rightarrow$

Wenn es regnet, dann ist die Straße nass.

- ▶ $W(\text{Es regnet.})=?$
- ▶ $W(\text{Die Straße ist nass.})=?$
- ▶ $W(\text{Wenn es regnet, dann ist die Straße nass.})=1$

$p \rightarrow q$ ist genau dann wahr,
wenn die Aussage p falsch oder die Aussage q wahr ist.

$W(p)$	$W(q)$	$W(p \rightarrow q)$
0	0	1
0	1	1
1	0	0
1	1	1

$$W(p \rightarrow q) = \begin{cases} 1 & \text{falls } W(p) \leq W(q) \\ 0 & \text{sonst} \end{cases}$$

Äquivalenz $\leftrightarrow$

$3 < 5$ gilt **genau dann, wenn** $0 < 5 - 3$ gilt.

▶ $W(3 < 5) = 1$

▶ $W(0 < 5 - 3) = 1$

▶ $W(3 < 5 \text{ gilt genau dann, wenn } 0 < 5 - 3 \text{ gilt.}) = 1$

$p \leftrightarrow q$ ist genau dann wahr, wenn
entweder **beide Aussagen p und q gelten**
oder **beide nicht gelten.**

$W(p)$	$W(q)$	$W(p \leftrightarrow q)$
0	0	1
0	1	0
1	0	0
1	1	1

$$W(p \leftrightarrow q) = \begin{cases} 1 & \text{falls } W(p) = W(q) \\ 0 & \text{sonst} \end{cases}$$

Was bisher geschah

- ▶ Inhalt und Organisation der Lehrveranstaltung
- ▶ Modellierung:
 - ▶ Diagramm: Realität – (formales) Modell (Aufgabe, Kontext, Lösung)
 - ▶ Einordnung in Prozess zur (maschinellen) Lösung von Aufgaben
 - ▶ Syntax und Semantik formaler Darstellungen
- ▶ Beispiel Münzenspiel: Modellierung von
 - ▶ Daten: Spielzustände
 - ▶ Abläufen:
 - ▶ Spielzug (Zustandsübergang)
 - ▶ graphische Darstellung gesamter Spiele
 - ▶ Eigenschaften: Endzustände (ohne Zugmöglichkeit)
- ▶ (klassische) Aussagenlogik

WH: Aussagenlogik

- ▶ Aussage = Behauptung
- ▶ Aussagenvariablen p (Syntax)
- ▶ klassische Logik: Wahrheitswerte $W(p) \in \{0, 1\}$ (Semantik)
- ▶ Kombination von Formeln durch Junktoren (Syntax)
- ▶ Wahrheitswert einer zusammengesetzten Formel wird aus Wahrheitswerten der Teilformeln berechnet. (Semantik)

		Syntax	Semantik
	Stelligkeit	Symbol	Wahrheitswertfunktion
wahr	0	$\top$	1
falsch	0	$\bot$	0
Konjunktion	2	$\wedge$	min
Disjunktion	2	$\vee$	max
Negation	1	$\neg$	$x \mapsto 1 - x$
Implikation	2	$\rightarrow$	$\leq$
Äquivalenz	2	$\leftrightarrow$	$=$

Modellierung in Aussagenlogik (Beispiele)

- ▶ Wenn Leo ein Bier bestellt, bestellt auch Mark eins.

Aussagen:

- ▶ Leo bestellt Bier.
- ▶ Mark bestellt Bier.

Aussagenvariablen:

- ▶ lb - Leo bestellt Bier.
- ▶ mb - Mark bestellt Bier.

Formel: $lb \rightarrow mb$

- ▶ Leo und Mark bestellen nie zugleich Bier.

Formel: $\neg(lb \wedge mb)$

- ▶ Mark bestellt höchstens dann Bier, wenn Leo Bier bestellt.

Formel: $\neg lb \rightarrow \neg mb$

- ▶ Wenn es morgens vor 7 oder abends nach 7 ist, ist es dunkel.

Aussagenvariablen:

- ▶ m - Es ist morgens vor 7.
- ▶ a - Es ist abends nach 7.
- ▶ d - Es ist dunkel.

Formel: $(m \vee a) \rightarrow d$

- ▶ Morgens vor 7 und abends nach 7 ist es dunkel.

Formel: $(m \rightarrow d) \wedge (a \rightarrow d)$

Aussagenlogische Formeln (Syntax)

Junktoren z.B. $\top, \bot$ (nullstellig), $\neg$ (einstellig),
 $\wedge, \vee, \rightarrow, \leftrightarrow$ (zweistellig)

Aussagenvariablen (Atome), z.B. $p, q, r, s, \dots$ oder $p_1, p_2, \dots$

Definition (induktiv)

Die Menge $AL(P)$ aller **aussagenlogischen Formeln** mit Aussagenvariablen aus der Menge P ist definiert durch:

IA: Alle Aussagenvariablen $p \in P$ sind Formeln. ($P \subseteq AL(P)$).

IS:

- ▶ $\top$ und $\bot$ sind Formeln.
- ▶ Ist φ eine Formel, dann ist auch $\neg\varphi$ eine Formel.
- ▶ Sind φ und ψ Formeln, dann sind auch $\varphi \vee \psi$, $\varphi \wedge \psi$, $\varphi \rightarrow \psi$ und $\varphi \leftrightarrow \psi$ Formeln.

IS': (Verallgemeinerung aller Unterpunkte zu IS)

Sind j ein n -stelliger Junktor und $\varphi_1, \dots, \varphi_n$ Formeln, dann ist auch $j(\varphi_1, \dots, \varphi_n)$ eine Formel.

(Aus $\{\varphi_1, \dots, \varphi_n\} \subseteq AL(P)$ folgt $j(\varphi_1, \dots, \varphi_n) \in AL(P)$.)

Aussagenlogische Formeln (Beispiele)

Junktoren der klassischen Aussagenlogik: $\{\top, \text{f}, \neg, \vee, \wedge, \rightarrow, \leftrightarrow\}$

kürzere Notation:

ohne äußere Klammern und Klammern um $\neg\varphi$

Beispiele:

$\top \wedge (\neg\top)$	Formel ohne Aussagenvariablen ($\in \text{AL}(\emptyset)$)
$\neg\neg\neg p$	Formel mit Aussagenvariable p ($\in \text{AL}(\{p\})$)
$\wedge(p \vee q)$	keine Formel (syntaktisch unkorrekt) ($\notin \text{AL}(P)$)
$\neg(p \rightarrow q)$	Formel mit Aussagenvariablen p, q ($\in \text{AL}(\{p, q\})$)
$\rightarrow q$	keine Formel (syntaktisch unkorrekt) ($\notin \text{AL}(P)$)

Baumstruktur (analog arithmetischen Termen)

Beispiel:

$$\varphi = ((p \wedge \neg q) \rightarrow (\neg r \vee (p \leftrightarrow q))) \quad \in \text{AL}(\{p, q, r\})$$

Menge aller Aussagenvariablen einer Formel

Definition (induktiv):

Für jede aussagenlogische Formel $\varphi \in AL(P)$ ist die Menge $\text{var}(\varphi)$ aller in φ vorkommenden Aussagenvariablen definiert durch:

IA: falls $\varphi = p$ (Atom), dann $\text{var}(\varphi) = \{p\}$

- IS:
- ▶ nullstellige Junktoren ($\top, \bot$):
für $\varphi = \top$ oder $\varphi = \bot$ gilt $\text{var}(\varphi) = \emptyset$
 - ▶ einstellige Junktoren ($\neg$):
für $\varphi = \neg\varphi_1$ gilt $\text{var}(\varphi) = \text{var}(\varphi_1)$
 - ▶ zweistellige Junktoren ($* \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$):
für $\varphi = \varphi_1 * \varphi_2$ gilt $\text{var}(\varphi) = \text{var}(\varphi_1) \cup \text{var}(\varphi_2)$

Beispiel:

Für $\varphi = p \rightarrow ((q \leftrightarrow \top) \vee (r \rightarrow q))$ gilt $\text{var}(\varphi) = \{p, q, r\}$

Anzahl der Variablenvorkommen in einer Formel

Definition (induktiv):

Für jede aussagenlogische Formel $\varphi \in \text{AL}(P)$ ist die Anzahl von Variablenvorkommen $\text{varcount}(\varphi)$ definiert durch:

IA: falls $\varphi = p$ (Atom), dann $\text{varcount}(\varphi) = 1$

IS: ▶ nullstellige Junktoren ($\top, \bot$):

$$\text{varcount}(\top) = \text{varcount}(\bot) = 0$$

▶ einstellige Junktoren ($\neg$):

$$\text{varcount}(\neg\varphi) = \text{varcount}(\varphi)$$

▶ zweistellige Junktoren ($* \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$):

$$\text{varcount}(\varphi * \psi) = \text{varcount}(\varphi) + \text{varcount}(\psi)$$

Beispiel:

$\varphi = p \rightarrow ((q \leftrightarrow \top) \vee (r \rightarrow q))$ gilt $\text{varcount}(\varphi) = 4$

$\text{varcount}(\varphi)$ ist die Anzahl aller mit Variablen markierten Blätter im Formelbaum von φ

Allgemein gilt $\text{varcount}(\varphi) \geq |\text{var}(\varphi)|$

Menge aller Teilformeln einer Formel

Definition (induktiv):

Für jede aussagenlogische Formel $\varphi \in \text{AL}(P)$ ist die Menge $\text{TF}(\varphi)$ aller in φ vorkommenden **Teilformeln** definiert durch:

IA: falls $\varphi = p$ (Atom), dann $\text{TF}(\varphi) = \{p\}$

IS: ► nullstellige Junktoren ($\top, \bot$):

für $\varphi = \top$ oder $\varphi = \bot$ gilt $\text{TF}(\varphi) = \{\varphi\}$

► einstellige Junktoren ($\neg$):

für $\varphi = \neg\varphi_1$ gilt $\text{TF}(\varphi) = \{\varphi\} \cup \text{TF}(\varphi_1)$

► zweistellige Junktoren ($* \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$):

für $\varphi = \varphi_1 * \varphi_2$ gilt $* \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$

$\text{TF}(\varphi) = \{\varphi\} \cup \text{TF}(\varphi_1) \cup \text{TF}(\varphi_2)$

Beispiel:

Für $\varphi = p \rightarrow ((q \leftrightarrow \top) \vee (r \rightarrow q))$ gilt

$$\text{TF}(\varphi) = \left\{ p \rightarrow ((q \leftrightarrow \top) \vee (r \rightarrow q)), p, (q \leftrightarrow \top) \vee (r \rightarrow q), \right. \\ \left. q \leftrightarrow \top, r \rightarrow q, q, \top, r \right\}$$

Prinzip der strukturellen Induktion

Beobachtung:

Bestimmung von Variablenmenge, Größe, Menge der Teilformeln einer aussagenlogischen Formel geschah nach demselben Schema:

Definition einer Funktion $f : AL(P) \rightarrow X$ durch

Induktion über die Struktur der Formel

Für jede aussagenlogische Formel $\varphi \in AL(P)$ ist der Funktionswert $f(\varphi)$ definiert durch:

IA: Definition des Funktionswertes $f(\varphi)$ für **Atome** $\varphi = p$
(Blätter im Formelbaum)

IS: Definition des Funktionswertes $f(\varphi)$ für **zusammengesetzte** Formeln φ durch die Funktionswerte der **Teilformeln** von φ

Aussagenlogische Interpretationen (Semantik)

(Belegungen der Aussagenvariablen mit Wahrheitswerten)

Interpretation (Belegung) für Formeln $\varphi \in \text{AL}(P)$:
ordnet jeder Aussagenvariable in P einen Wahrheitswert zu,
ist eine Funktion $W : P \longrightarrow \{0, 1\}$
(eine Zeile in der WW-Tabelle für φ)

Beispiel: $P = \{p, q\}$ und W_{10} mit $W_{10}(p) = 1$ und $W_{10}(q) = 0$

Wahrheitswerte für Formeln

Belegung $W : P \rightarrow \{0, 1\}$ fortgesetzt zu Funktion $W : \mathbf{AL}(P) \rightarrow \{0, 1\}$

Der Wert $W(\varphi)$ der Formel φ in der Interpretation W wird **induktiv** mit den Wahrheitswertfunktionen der Junktoren aus den Werten der Teilformeln von φ bestimmt:

IA: falls $\varphi = p$ (Atom), dann $W(\varphi) = W(p)$

IS: ▶ nullstellige Junktoren $\mathbb{t}, \mathbb{f}$: $W(\mathbb{t}) = 1$, $W(\mathbb{f}) = 0$

▶ einstelliger Junktor $\neg$:

für $\varphi = \neg\psi$ gilt $W(\neg\psi) = \llbracket \neg \rrbracket W(\psi) = 1 - W(\psi)$

▶ zweistellige Junktoren $*$ $\in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$:

$$W(\psi_1 \wedge \psi_2) = W(\psi_1) \llbracket \wedge \rrbracket W(\psi_2) = \min(W(\psi_1), W(\psi_2))$$

$$W(\psi_1 \vee \psi_2) = W(\psi_1) \llbracket \vee \rrbracket W(\psi_2) = \max(W(\psi_1), W(\psi_2))$$

$$W(\psi_1 \rightarrow \psi_2) = W(\psi_1) \llbracket \rightarrow \rrbracket W(\psi_2) = \begin{cases} 1 & \text{falls } W(\psi_1) \leq W(\psi_2) \\ 0 & \text{sonst} \end{cases}$$

$$W(\psi_1 \leftrightarrow \psi_2) = W(\psi_1) \llbracket \leftrightarrow \rrbracket W(\psi_2) = \begin{cases} 1 & \text{falls } W(\psi_1) = W(\psi_2) \\ 0 & \text{sonst} \end{cases}$$

Beispiel: $\varphi = ((p \wedge \neg q) \rightarrow (\neg r \vee (p \leftrightarrow q)))$

$W_{010}(p) = 0$, $W_{010}(q) = 1$, $W_{010}(r) = 0$, $W_{010}(\varphi) = \dots$

Wahrheitstabelle

Darstellung der Werte einer Formel $\varphi \in \text{AL}(P)$ in
allen möglichen Interpretationen $W \in \mathbb{W}(P)$ in einer Tabelle

Jede Zeile repräsentiert eine Interpretation $W : \text{var}(\varphi) \rightarrow \{0, 1\}$

$W(p_1)$	$W(p_2)$	$\dots$	$W(p_{n-1})$	$W(p_n)$	$W(\varphi)$
0	0	$\dots$	0	0	
0	0	$\dots$	0	1	
		$\vdots$			
1	1	$\dots$	1	1	

Beispiel: $((p \wedge \neg q) \rightarrow (\neg r \vee (p \leftrightarrow q)))$

Wahrheitstabelle von Formeln φ mit n Aussagenvariablen
sind Wertetabelle n -stelliger Boolescher Funktionen

$f_\varphi : \{0, 1\}^n \rightarrow \{0, 1\}$.

Die Semantik jeder aussagenlogischen Formel φ mit n
Aussagenvariablen ist eine n -stellige Boolesche Funktion.

Modelle aussagenlogischer Formeln

Die Belegung $W : P \rightarrow \{0, 1\}$ der Aussagenvariablen in P **erfüllt** die Formel $\varphi \in \text{AL}(P)$ (ist ein **Modell** von φ) genau dann, wenn $W(\varphi) = 1$.

Beispiel:

Modelle (erfüllende Belegungen) der Formel

$$\varphi = p \vee (q \wedge \neg p) \in \text{AL}(\{p, q\})$$

- ▶ Belegung $W_{10} : \{p, q\} \rightarrow \{0, 1\}$ mit $W_{10}(p) = 1$ und $W_{10}(q) = 0$ ist ein **Modell** für φ , weil $W_{10}(\varphi) = 1$.
- ▶ Belegung $W_{00} : \{p, q\} \rightarrow \{0, 1\}$ ist **kein Modell** für φ , weil $W_{00}(\varphi) = 0$.
- ▶ W_{01} mit $W_{01}(p) = 0$ und $W_{01}(q) = 1$ ist ein Modell für φ ,
- ▶ W_{11} mit $W_{11}(p) = 1$ und $W_{11}(q) = 1$ ist ein Modell für φ .

Modellmengen aussagenlogischer Formeln

Modellmenge (Menge aller Modelle) der Formel $\varphi \in \text{AL}(P)$:

$$\text{Mod}(\varphi) = \{W : P \rightarrow \{0, 1\} \mid W(\varphi) = 1\}$$

(Diese Darstellung ist oft kürzer als die WW-Tabelle)

Beispiele:

- ▶ $\text{Mod}(p \vee (q \wedge \neg p)) = \{W_{10}, W_{01}, W_{11}\},$
- ▶ $\text{Mod}(p \rightarrow p) = \{W_0, W_1\} = \{W : \{p\} \rightarrow \{0, 1\}\}$
(alle möglichen Belegungen für p),
- ▶ $\text{Mod}(p \wedge \neg p) = \emptyset$

Was bisher geschah: klassische Aussagenlogik

Syntax Symbole und Struktur

- ▶ Menge P von **Aussagenvariablen** $(p, q, r, \dots)$
- ▶ **Junktoren** (je mit Stelligkeit):
 $\top(0), \perp(0), \neg(1), \vee(2), \wedge(2), \rightarrow(2), \leftrightarrow(2)$
- ▶ **aussagenlogische Formeln** $AL(P)$ (induktive Def.):
 - IA Atome (Aussagenvariablen) $\in P$
 - IS zusammengesetzte Formeln $(\varphi, \psi, \eta, \dots)$:
Verknüpfung von Formeln durch Junktoren

Prinzip der strukturellen Induktion über **Baumstruktur** von Formeln

Semantik (Bedeutung der Syntaxelemente)

- ▶ einer **Aussagenvariablen**: **Wahrheitswert** $\in \{0, 1\}$
- ▶ aller Aussagenvariablen einer Menge P :
Belegung (Interpretation) $W : P \rightarrow \{0, 1\}$
- ▶ eines n -stelligem **Junktors** $*$:
Wahrheitswertfunktion $\llbracket * \rrbracket : \{0, 1\}^n \rightarrow \{0, 1\}$
- ▶ einer Formel unter einer Belegung W :
Funktion $W : AL(P) \rightarrow \{0, 1\}$
- ▶ **Modell** (erfüllende Belegung) für $\varphi \in AL(P)$:
Belegung $W : P \rightarrow \{0, 1\}$ mit $W(\varphi) = 1$
- ▶ **Modellmenge** $\text{Mod}(\varphi)$ der Formel $\varphi \in AL(P)$:
Menge aller Modelle von φ

WH: Modelle aussagenlogischer Formeln

Die Belegung $W : P \rightarrow \{0, 1\}$ der Aussagenvariablen in P **erfüllt** die Formel $\varphi \in \text{AL}(P)$ (ist ein **Modell** von φ) genau dann, wenn $W(\varphi) = 1$.

Modellmenge (Menge aller Modelle) der Formel $\varphi \in \text{AL}(P)$:

$$\text{Mod}(\varphi) = \{W : P \rightarrow \{0, 1\} \mid W(\varphi) = 1\}$$

(Diese Darstellung ist oft kürzer als die WW-Tabelle)

Beispiele:

- ▶ $\text{Mod}(p \vee (q \wedge \neg p)) = \{W_{10}, W_{01}, W_{11}\},$
- ▶ $\text{Mod}(p \rightarrow p) = \{W_0, W_1\} = \{W : \{p\} \rightarrow \{0, 1\}\}$
(alle möglichen Belegungen für p),
- ▶ $\text{Mod}(p \wedge \neg p) = \emptyset$

Modellierungsbeispiel Party

umgangssprachliche Beschreibung der Situation (Kontext):

Anna geht zur Party, wenn Max oder Paul hingehen. (1)

Max geht zur Party, wenn Paul nicht hinget. (2)

Anna geht nirgends ohne ihren Hund hin. (3)

formale Beschreibung der Situation:

► elementare Aussagen (Atome):

a Anna geht zur Party.

m Max geht zur Party.

p Paul geht zur Party.

h Annas Hund geht zur Party.

$P = \{a, h, m, p\}$ (Menge der Aussagenvariablen)

► zusammengesetzte Aussage (aussagenlogische **Formel**):

$$\varphi = \underbrace{((m \vee p) \rightarrow a)}_{(1)} \wedge \underbrace{(\neg p \rightarrow m)}_{(2)} \wedge \underbrace{\neg(a \wedge \neg h)}_{(3)} \in \text{AL}(P)$$

Modellmenge $\text{Mod}(\varphi) = \dots$

Menge aller Situationen, in denen alle Aussagen wahr sind

Welche dieser Situationen ist Hunde-frei? $\text{Mod}(\varphi \wedge \neg h) = \dots$

Erfüllbarkeit und Allgemeingültigkeit von Formeln

Definition: Eine Formel $\varphi \in \text{AL}(P)$ heißt

erfüllbar , wenn $\text{Mod}(\varphi) \neq \emptyset$,

also (wenigstens) eine Belegung $W : P \rightarrow \{0, 1\}$ mit $W(\varphi) = 1$ existiert

Beispiel: $\neg p \rightarrow p$

unerfüllbar (Widerspruch), wenn $\text{Mod}(\varphi) = \emptyset$,

also keine Belegung $W : P \rightarrow \{0, 1\}$ mit $W(\varphi) = 1$ existiert.

(wenn also für jede Belegung W gilt $W(\varphi) = 0$),

Beispiel: $p \wedge \neg p$

allgemeingültig (Tautologie), wenn $\text{Mod}(\varphi) = \{W : P \rightarrow \{0, 1\}\}$,

also für jede Belegung $W : P \rightarrow \{0, 1\}$ gilt $W(\varphi) = 1$

(wenn also keine Belegung W mit $W(\varphi) = 0$ existiert).

Beispiel: $p \rightarrow p$

Fakt

Eine Formel $\varphi \in \text{AL}(P)$ ist genau dann allgemeingültig, wenn die Formel $\neg\varphi$ unerfüllbar ist.

Beweis: ...

Formelmengen

Formelmenge $\Phi \subseteq \text{AL}(P)$

(Menge von Bedingungen)

Beispiele:

- ▶ $\{p, p \rightarrow q\} \subseteq \text{AL}(P)$
- ▶ $\{p, p \rightarrow q, \neg q\} \subseteq \text{AL}(P)$
- ▶ $\{p \rightarrow q\} \subseteq \text{AL}(P)$
- ▶ $\emptyset \subseteq \text{AL}(P)$

Semantik von Formelmengen

Eine Belegung $W : P \rightarrow \{0, 1\}$ erfüllt eine Menge $\Phi \subseteq \text{AL}(P)$ von Formeln genau dann, wenn W **jede** Formel $\varphi \in \Phi$ erfüllt.

Beispiele:

- ▶ einziges Modell für $\{p, p \rightarrow q\}$: W_{11}
- ▶ $\{p, p \rightarrow q, \neg q\}$ hat kein Modell,
- ▶ Modelle für $\{p \rightarrow q\}$: W_{00}, W_{01}, W_{11}
- ▶ Jede Belegung ist ein Modell für die Formelmenge $\emptyset$.

Modellmengen von Formelmengen

Menge aller Modelle einer Menge $\Phi \subseteq \text{AL}(P)$ von Formeln:

$$\text{Mod}(\Phi) = \{W : P \rightarrow \{0, 1\} \mid \text{für jedes } \psi \in \Phi \text{ gilt } W \in \text{Mod}(\psi)\}$$

kürzere Formulierung derselben Definition:

$$\text{Mod}(\Phi) = \bigcap_{\psi \in \Phi} \text{Mod}(\psi)$$

Beispiele:

- ▶ $\text{Mod}(\{p, p \rightarrow q\}) = \text{Mod}(\{p\}) \cap \text{Mod}(\{p \rightarrow q\})$
 $= \{W_{10}, W_{11}\} \cap \{W_{00}, W_{01}, W_{11}\} = \{W_{11}\},$
- ▶ $\text{Mod}(\{p, p \rightarrow q, \neg q\}) = \emptyset,$
- ▶ $\text{Mod}(\{p \rightarrow q\}) = \{W_{00}, W_{01}, W_{11}\}$
- ▶ $\text{Mod}(\emptyset) = \{W : P \rightarrow \{0, 1\}\}$ (Menge aller Belegungen)

Fakt

Eine Belegung $W : P \rightarrow \{0, 1\}$ erfüllt eine endliche Formelmenge

$\Phi = \{\varphi_1, \dots, \varphi_n\}$ genau dann, wenn sie die Formel $\varphi_1 \wedge \dots \wedge \varphi_n$ erfüllt.

kürzere Formulierung derselben Aussage:

$$\text{Mod}(\Phi) = \text{Mod} \left(\bigwedge_{\psi \in \Phi} \psi \right)$$

Modellierung durch aussagenlogische Formelmengen

Aussagen:

1. Es wird nicht mehr viel Eis gekauft, wenn es kalt ist.
2. Der Eisverkäufer ist traurig, wenn nicht viel Eis gekauft wird.
3. Es ist kalt.

Darstellung als Formelmenge $\Phi \subseteq \text{AL}(\{k, t, v\})$:

$$\Phi = \{k \rightarrow \neg v, \neg v \rightarrow t, k\}$$

$$\text{Mod}(\Phi) = \{W_{110}\}$$

neue zusätzliche Aussage:

4. Der Eisverkäufer ist nicht traurig.

Erweiterung der Formelmenge Φ zu

$$\Phi' = \Phi \cup \{\neg t\} = \{k \rightarrow \neg v, \neg v \rightarrow t, k, \neg t\}$$

$$\text{Mod}(\Phi') = \emptyset$$

(Formelmenge Φ' unerfüllbar)

Semantische Äquivalenz aussagenlogischer Formeln

Definition

Zwei Formeln $\varphi, \psi \in \text{AL}(P)$ heißen genau dann
(semantisch) äquivalent ($\varphi \equiv \psi$), wenn $\text{Mod}(\varphi) = \text{Mod}(\psi)$.

alternative Formulierung:

$\varphi \equiv \psi$ gdw. für jede Belegung $W : P \rightarrow \{0, 1\}$ gilt $W(\varphi) = W(\psi)$.

Äquivalente Formeln haben dieselbe Semantik (Wahrheitswertfunktion).

Beispiele: $p \rightarrow q \equiv \neg p \vee q$, $p \vee q \equiv \neg p \rightarrow q$

allgemein: Für alle Formeln $\varphi, \psi \in \text{AL}(P)$ gilt

$$\varphi \rightarrow \psi \equiv \neg \varphi \vee \psi$$

$$\varphi \vee \psi \equiv \neg \varphi \rightarrow \psi$$

$$\varphi \wedge \psi \equiv \neg(\varphi \rightarrow \neg \psi)$$

$$\varphi \leftrightarrow \psi \equiv (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$$

Achtung: Das Symbol $\equiv$ ist kein Junktor (Syntax), sondern ein
Symbol für eine Beziehung zwischen Formeln (Semantik).

Aber: Für alle Formeln $\varphi, \psi \in \text{AL}(P)$ lässt sich beweisen:

$\varphi \equiv \psi$ gilt genau dann, wenn die Formel $\varphi \leftrightarrow \psi$ allgemeingültig ist.

Was bisher geschah: klassische Aussagenlogik

Syntax Symbole und Struktur, **Junktoren**: $\top, \bot, \neg, \vee, \wedge, \rightarrow, \leftrightarrow$

Prinzip der strukturellen Induktion über **Baumstruktur** von Formeln, arithmetischen Ausdrücken usw.

- ▶ induktive Definition von (unendlichen) **Mengen**
- ▶ induktive Definition von **Funktionen** auf induktiv definierten Mengen

Semantik (Bedeutung der Syntaxelemente)

- ▶ eines Junktors: Wahrheitswertfunktion
- ▶ einer Aussagenvariablen: Wahrheitswert
- ▶ einer Formel aus $AL(P)$ unter einer Belegung
 $W : P \rightarrow \{0, 1\}$: Funktion $W : AL(P) \rightarrow \{0, 1\}$
- ▶ einer Formel aus $AL(P)$ unter allen möglichen Belegungen:
Boolesche Funktion
- ▶ **Modelle** (erfüllende Belegungen) von Formeln
- ▶ **Modellmengen** aussagenlogischer Formeln
- ▶ **Erfüllbarkeit, Allgemeingültigkeit** von Formeln
- ▶ **semantische Äquivalenz** von Formeln

WH: Semantische Äquivalenz aussagenlogischer Formeln

Definition

Zwei Formeln $\varphi, \psi \in \text{AL}(P)$ heißen genau dann
(semantisch) äquivalent ($\varphi \equiv \psi$), wenn $\text{Mod}(\varphi) = \text{Mod}(\psi)$.

alternative Formulierung:

$\varphi \equiv \psi$ gdw. für jede Belegung $W : P \rightarrow \{0, 1\}$ gilt $W(\varphi) = W(\psi)$.

Äquivalente Formeln haben dieselbe Semantik (Wahrheitswertfunktion).

Beispiele: $p \rightarrow q \equiv \neg p \vee q$, $p \vee q \equiv \neg p \rightarrow q$

allgemein: Für alle Formeln $\varphi, \psi \in \text{AL}(P)$ gilt

$$\varphi \rightarrow \psi \equiv \neg \varphi \vee \psi$$

$$\varphi \vee \psi \equiv \neg \varphi \rightarrow \psi$$

$$\varphi \wedge \psi \equiv \neg(\varphi \rightarrow \neg \psi)$$

$$\varphi \leftrightarrow \psi \equiv (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$$

Achtung: Das Symbol $\equiv$ ist kein Junktor (Syntax), sondern ein
Symbol für eine Beziehung zwischen Formeln (Semantik).

Aber: Für alle Formeln $\varphi, \psi \in \text{AL}(P)$ lässt sich beweisen:

$\varphi \equiv \psi$ gilt genau dann, wenn die Formel $\varphi \leftrightarrow \psi$ allgemeingültig ist.

Nachweis von Aussagen über alle Formeln

Aussagen der Form: Für **alle** Formeln $\varphi, \psi \in \text{AL}(P)$ gilt ...
lassen sich **nicht** mit Wahrheitstabelle nachweisen.

Beispiel:

Für alle Formeln $\varphi, \psi \in \text{AL}(P)$ gilt $\varphi \rightarrow \psi \equiv \neg\varphi \vee \psi$

Nach Definition von $\equiv$ ist zu zeigen: **Mod**($\varphi \rightarrow \psi$) = **Mod**($\neg\varphi \vee \psi$)

$$\begin{aligned} \text{Mod}(\varphi \rightarrow \psi) &\stackrel{\text{Def. Mod}}{=} \{W : P \rightarrow \{0, 1\} \mid W(\varphi \rightarrow \psi) = 1\} \\ &\stackrel{\text{Def. } \llbracket \rightarrow \rrbracket}{=} \{W : P \rightarrow \{0, 1\} \mid W(\varphi) \leq W(\psi)\} \\ &\stackrel{\text{Def. } \leq}{=} \{W : P \rightarrow \{0, 1\} \mid W(\varphi) = 0 \text{ oder } W(\psi) = 1\} \\ &\stackrel{\text{Def. } \llbracket \neg \rrbracket}{=} \{W : P \rightarrow \{0, 1\} \mid W(\neg\varphi) = 1 \text{ oder } W(\psi) = 1\} \\ &\stackrel{\text{Def. max}}{=} \{W : P \rightarrow \{0, 1\} \mid \max(W(\neg\varphi), W(\psi)) = 1\} \\ &\stackrel{\text{Def. } \llbracket \vee \rrbracket}{=} \{W : P \rightarrow \{0, 1\} \mid W(\neg\varphi \vee \psi) = 1\} \\ &\stackrel{\text{Def. Mod}}{=} \text{Mod}(\neg\varphi \vee \psi) \end{aligned}$$

Wichtige Äquivalenzen

Für alle aussagenlogischen Formeln $\varphi, \psi, \eta \in \text{AL}(P)$ gilt:

- ▶ $\varphi \vee \varphi \equiv \varphi, \quad \varphi \wedge \varphi \equiv \varphi, \quad \varphi \vee \text{ff} \equiv \varphi, \quad \varphi \wedge \text{tt} \equiv \varphi$
- ▶ $\varphi \vee \psi \equiv \psi \vee \varphi, \quad \varphi \wedge \psi \equiv \psi \wedge \varphi$
(Kommutativität von $\wedge$ und $\vee$)
- ▶ $\varphi \vee (\psi \vee \eta) \equiv (\varphi \vee \psi) \vee \eta$
 $\varphi \wedge (\psi \wedge \eta) \equiv (\varphi \wedge \psi) \wedge \eta$
(Assoziativität von $\wedge$ und $\vee$)
- ▶ $\varphi \wedge (\psi \vee \eta) \equiv (\varphi \wedge \psi) \vee (\varphi \wedge \eta)$
 $\varphi \vee (\psi \wedge \eta) \equiv (\varphi \vee \psi) \wedge (\varphi \vee \eta)$
(Distributivgesetze)
- ▶ $\neg\neg\varphi \equiv \varphi$ (Doppelnegation)
- ▶ $\neg(\varphi \vee \psi) \equiv \neg\varphi \wedge \neg\psi, \quad \neg(\varphi \wedge \psi) \equiv \neg\varphi \vee \neg\psi$
(DeMorgansche Regeln)
- ▶ $\varphi \vee \psi \equiv \neg(\neg\varphi \wedge \neg\psi), \quad \varphi \wedge \psi \equiv \neg(\neg\varphi \vee \neg\psi)$
(Dualität von $\wedge$ und $\vee$)
- ▶ $\varphi \rightarrow \psi \equiv \neg\psi \rightarrow \neg\varphi$ (Kontraposition)
- ▶ $(\varphi \wedge \psi) \vee (\neg\varphi \wedge \psi) \equiv \psi$ (Fallunterscheidung)

WH: Strukturelle Induktion

Induktive Definition von Mengen

Beispiel: Menge $AL_{\{\neg, \rightarrow\}}(\{p, q\})$ aller aussagenlogischen Formeln, die nur die Aussagenvariablen p, q und die Junktoren $\neg$ und $\rightarrow$ enthalten

IA: Grundbausteine, im Bsp.: elementare Formeln (Atome) p, q

IS: Regeln zur Konstruktion zusammengesetzter Elemente

im Bsp.: Zusammensetzen von Formeln durch Junktoren

Für alle Formeln φ_1 und φ_2 aus der Menge $AL_{\{\neg, \rightarrow\}}(\{p, q\})$ sind auch $\neg\varphi_1$ und $\varphi_1 \rightarrow \varphi_2$ in der Menge $AL_{\{\neg, \rightarrow\}}(\{p, q\})$.

Beispiel: $\neg p \rightarrow (\neg q \rightarrow p) \in AL_{\{\neg, \rightarrow\}}(\{p, q\})$, $\neg p \vee q \notin AL_{\{\neg, \rightarrow\}}(\{p, q\})$

ermöglicht induktive Definition von Funktionen auf induktiv definierten Mengen (Beispiel: Funktion $f : AL_{\{\neg, \rightarrow\}}(\{p, q\}) \rightarrow \mathbb{Z}$):

IA: Funktionswert für Grundbausteine, z.B. $f(p) = 2$, $f(q) = 3$

IS: Vorschrift zur Berechnung des Funktionswertes des zusammengesetzten Elementes aus Funktionswerten der Teilstrukturen

z.B. $f(\neg\varphi_1) = 2f(\varphi_1)$, $f(\varphi_1 \rightarrow \varphi_2) = f(\varphi_1) + 3f(\varphi_2)$

Beispiel: $f(\neg p \rightarrow (\neg q \rightarrow p)) = 40$

Beweise durch strukturelle Induktion

Induktiver **Nachweis von Eigenschaften** (E) jedes Elementes induktiv definierter Mengen

Beispiel: Für jede Formel $\varphi \in \text{AL}_{\{\neg, \rightarrow\}}(\{p\})$ gilt $f(\varphi) > \text{varcount}(\varphi)$ (E)

IA: Nachweis der Eigenschaft (E) für Grundbausteine,

im Bsp.: $f(p) \stackrel{(\text{Def. } f)}{=} 2 > 1 \stackrel{(\text{Def. varcount})}{=} \text{varcount}(p)$

IS: Nachweis der Eigenschaft (E) für zusammengesetzte Elemente aus den Nachweisen der Eigenschaft für die Teilstrukturen

IH (Induktionshypothese): Eigenschaft (E) für Teilstrukturen
im Bsp.: $f(\varphi_1) > \text{varcount}(\varphi_1)$ und $f(\varphi_2) > \text{varcount}(\varphi_2)$

IB (Induktionsbehauptung): Eigenschaft (E) für die (aus diesen Teilstrukturen) zusammengesetzten Elemente, im Bsp.:

$f(\neg\varphi_1) > \text{varcount}(\neg\varphi_1)$, $f(\varphi_1 \rightarrow \varphi_2) > \text{varcount}(\varphi_1 \rightarrow \varphi_2)$

B (Induktionsbeweis): Nachweis, dass IB aus IH folgt, Bsp.:

$$f(\neg\varphi_1) = 2f(\varphi_1) \stackrel{(\text{IH})}{>} 2\text{varcount}(\varphi_1) \geq \text{varcount}(\varphi_1) = \text{varcount}(\neg\varphi_1)$$

$$\begin{aligned} f(\varphi_1 \rightarrow \varphi_2) &= f(\varphi_1) + 3f(\varphi_2) \stackrel{(\text{IH})}{>} \text{varcount}(\varphi_1) + 3\text{varcount}(\varphi_2) \\ &\geq \text{varcount}(\varphi_1) + \text{varcount}(\varphi_2) = \text{varcount}(\varphi_1 \rightarrow \varphi_2) \end{aligned}$$

Junktorbasen (vollständige Operatorensysteme)

Zu einer Menge J von Junktoren ist die Menge $AL_J(P)$ definiert durch

IA: Für jede Aussagenvariable $p \in P$ gilt $p \in AL_J(P)$

- IS:
- ▶ für jeden 0-stelligen Junktor $*$ $\in J$ gilt $*$ $\in AL_J(P)$
 - ▶ für jeden 1-stelligen Junktor $*$ $\in J$ und alle Formeln $\varphi \in AL_J(P)$ gilt $*\varphi \in AL_J(P)$
 - ▶ für jeden 2-stelligen Junktor $*$ $\in J$ und alle Formeln $\varphi, \psi \in AL_J(P)$ gilt $\varphi * \psi \in AL_J(P)$

Definition: Eine Menge J von Junktoren heißt genau dann **Junktorbasis** (vollständiges Operatorensystem), wenn zu jeder Formel $\varphi \in AL(P)$ eine Formel $\psi \in AL_J(P)$ mit $\varphi \equiv \psi$ existiert.

Beispiele:

- ▶ Die Mengen $\{\neg, \vee, \wedge\}$, $\{\neg, \vee\}$, $\{\neg, \wedge\}$ sind Junktorbasen.
- ▶ Die Mengen $\{\neg, \rightarrow\}$, $\{\text{ff}, \rightarrow\}$ sind Junktorbasen. (ÜA)
- ▶ Die Mengen $\{\vee, \wedge\}$ und $\{\vee, \wedge, \rightarrow\}$ sind keine Junktorbasen. (ÜA)

Beweis durch strukturelle Induktion – Beispiel

Fakt: Die Menge $\{\neg, \vee, \wedge\}$ ist eine Junktorbasis.

alternative Formulierung:

Zu jeder Formel $\varphi \in \text{AL}(P)$ existiert eine Formel ψ mit den Eigenschaften

E1 $\psi \in \text{AL}_{\{\neg, \vee, \wedge\}}(P)$ und

E2 $\varphi \equiv \psi$ (d.h. $\text{Mod}(\varphi) = \text{Mod}(\psi)$)

Beweis: induktiv über die Struktur von $\varphi \in \text{AL}(P)$

(Konstruktion einer Formel ψ mit den Eigenschaften E1 und E2):

Induktionsanfang: $\varphi = p \in P$

Zu jedem $\varphi = p \in P$ erfüllt $\psi = p$ (Ansatz) beide Eigenschaften

E1 $\psi \in \text{AL}_{\{\neg, \vee, \wedge\}}(P)$, nach IA in der Def. von $\text{AL}_{\{\neg, \vee, \wedge\}}(P)$ und

E2 $\varphi \equiv \psi$, wegen $\text{Mod}(\varphi) = \text{Mod}(p) = \text{Mod}(\psi)$

Beweis durch strukturelle Induktion – IH und IB

Induktionsschritt: φ ist eine zusammengesetzte Formel

IH : Zu $\varphi_1, \varphi_2 \in \text{AL}(P)$ existieren Formeln ψ_1, ψ_2 mit

E1 $\psi_1, \psi_2 \in \text{AL}_{\{\neg, \vee, \wedge\}}(P)$ und

E2 $\varphi_1 \equiv \psi_1$ und $\varphi_2 \equiv \psi_2$

IB : für jede mögliche Struktur (Junktor in der Wurzel) von φ

IB $\neg$: zu $\varphi = \neg\varphi_1$ existiert eine Formel ψ mit

E1 $\psi \in \text{AL}_{\{\neg, \vee, \wedge\}}(P)$ und

E2 $\varphi \equiv \psi$

IB $\vee$: zu $\varphi = \varphi_1 \vee \varphi_2$ existiert eine Formel ψ mit E1 und E2

IB $\wedge$: zu $\varphi = \varphi_1 \wedge \varphi_2$ existiert eine Formel ψ mit E1 und E2

IB $\rightarrow$: zu $\varphi = \varphi_1 \rightarrow \varphi_2$ existiert eine Formel ψ mit E1 und E2

IB $\leftrightarrow$: zu $\varphi = \varphi_1 \leftrightarrow \varphi_2$ existiert eine Formel ψ mit E1 und E2

IB $\mathbb{t}$: zu $\varphi = \mathbb{t}$ existiert eine Formel ψ mit E1 und E2

IB $\mathbb{f}$: zu $\varphi = \mathbb{f}$ existiert eine Formel ψ mit E1 und E2

Beweis durch strukturelle Induktion – (einfache) Beweise

Induktionsschritt: Beweise

(Schritt B für jede mögliche Struktur der Formel $\varphi \in \text{AL}(P)$)

B $\neg$: z.z.: aus IH folgt $\text{IB}\neg$, **Ansatz:** $\psi = \neg\psi_1$

Beweis: Für $\psi = \neg\psi_1$ gelten

E1 $\psi \in \text{AL}_{\{\neg, \vee, \wedge\}}(P)$ gilt wegen $\psi_1 \in \text{AL}_{\{\neg, \vee, \wedge\}}(P)$ (nach IH) und IS (für $\neg$) in der Definition von $\text{AL}_{\{\neg, \vee, \wedge\}}(P)$

E2 $\varphi \equiv \psi$ (wird gezeigt durch $\text{Mod}(\psi) = \text{Mod}(\varphi)$)

$$\begin{array}{lll} \text{Mod}(\psi) & \stackrel{\text{Def. } \psi}{=} & \text{Mod}(\neg\psi_1) \\ & \stackrel{\text{Def. Mod}}{=} & \{W : P \rightarrow \{0, 1\} \mid W(\neg\psi_1) = 1\} \\ & \stackrel{\text{Def. } \llbracket \neg \rrbracket}{=} & \{W : P \rightarrow \{0, 1\} \mid W(\psi_1) = 0\} \\ \text{IH: } \psi_1 \equiv \varphi_1 & \stackrel{\text{IH}}{=} & \{W : P \rightarrow \{0, 1\} \mid W(\varphi_1) = 0\} \\ & \stackrel{\text{Def. } \llbracket \neg \rrbracket}{=} & \{W : P \rightarrow \{0, 1\} \mid W(\neg\varphi_1) = 1\} \\ & \stackrel{\text{Def. Mod}}{=} & \text{Mod}(\neg\varphi_1) \stackrel{\text{Struktur von } \varphi}{=} \text{Mod}(\varphi) \end{array}$$

B $\vee$: z.z.: aus IH folgt $\text{IB}\vee$, **Ansatz:** $\psi = \psi_1 \vee \psi_2$

Beweis: Für $\psi = \psi_1 \vee \psi_2$ gelten E1 wegen ..., E2 wegen ...

B $\wedge$: z.z.: aus IH folgt $\text{IB}\wedge$, **Ansatz:** $\psi = \psi_1 \wedge \psi_2$, Beweis : ...

Beweis durch strukturelle Induktion – Beweise

$B \rightarrow$: z.z.: aus IH folgt $IB \rightarrow$, **Ansatz**: $\psi = \neg\psi_1 \vee \psi_2$

Beweis: Für $\psi = \neg\psi_1 \vee \psi_2$ gelten

E1 $\psi \in \text{AL}_{\{\neg, \vee, \wedge\}}(P)$ wegen ...

E2 $\varphi \equiv \psi$ (gezeigt durch $\text{Mod}(\psi) = \text{Mod}(\varphi)$)

$$\begin{aligned} \text{Mod}(\psi) &\stackrel{\text{Def. } \psi}{=} \text{Mod}(\neg\psi_1 \vee \psi_2) \\ &\stackrel{\text{Def. Mod}}{=} \{W : P \rightarrow \{0, 1\} \mid W(\neg\psi_1 \vee \psi_2) = 1\} \\ &\stackrel{\text{Def. } \llbracket \vee \rrbracket}{=} \{W : P \rightarrow \{0, 1\} \mid \max(W(\neg\psi_1), W(\psi_2)) = 1\} \\ &\stackrel{\text{Def. max}}{=} \{W : P \rightarrow \{0, 1\} \mid W(\neg\psi_1) = 1 \text{ oder } W(\psi_2) = 1\} \\ &\stackrel{\text{Def. } \llbracket \neg \rrbracket}{=} \{W : P \rightarrow \{0, 1\} \mid W(\psi_1) = 0 \text{ oder } W(\psi_2) = 1\} \\ &\stackrel{\text{IH}}{=} \{W : P \rightarrow \{0, 1\} \mid W(\varphi_1) = 0 \text{ oder } W(\varphi_2) = 1\} \\ &\stackrel{\text{Def. } \leq}{=} \{W : P \rightarrow \{0, 1\} \mid W(\varphi_1) \leq W(\varphi_2)\} \\ &\stackrel{\text{Def. } \llbracket \rightarrow \rrbracket}{=} \{W : P \rightarrow \{0, 1\} \mid W(\varphi_1 \rightarrow \varphi_2) = 1\} \\ &\stackrel{\text{Def. Mod}}{=} \text{Mod}(\varphi_1 \rightarrow \varphi_2) \stackrel{\text{Struktur von } \varphi}{=} \text{Mod}(\varphi) \end{aligned}$$

analog: $B \leftrightarrow$, $B \dagger$ (Ansatz: $p \vee \neg p$ für ein beliebiges $p \in P$), $B \text{f}$

Umformen von Formeln

Nebenwirkung des (induktiven) Beweises zur
Junktorbasis-Eigenschaft:

Fakt:

Jede aussagenlogische Formel kann schrittweise durch Ersetzung
äquivalenter Teilformeln in semantisch äquivalente Formeln
umgeformt werden.

(Änderung der Syntax bei unveränderter Semantik)

Beispiel:

$$\begin{array}{lll} \neg(p \rightarrow q) & \begin{array}{l} (\varphi \rightarrow \psi \equiv \neg\varphi \vee \psi) \\ \equiv \end{array} & \neg(\neg p \vee q) \\ & \begin{array}{l} \text{(deMorgan)} \\ \equiv \end{array} & \neg\neg p \wedge \neg q \\ & \begin{array}{l} (\neg\neg\varphi \equiv \varphi) \\ \equiv \end{array} & p \wedge \neg q \end{array}$$

Was bisher geschah

Modellierung von **Aussagen** in (klassischer) Aussagenlogik

Syntax:

- ▶ Atome : Aussagenvariablen
- ▶ Junktoren $\neg, \vee, \wedge, \rightarrow, \leftrightarrow$
- ▶ induktive Definitionen von Formelmengen: $AL(P), AL_J(P)$
- ▶ Baumstruktur von Formeln
- ▶ Beweise durch strukturelle Induktion
- ▶ Junktorbasen
- ▶ äquivalente Umformungen

Semantik:

- ▶ Belegungen der Aussagenvariablen
- ▶ Wahrheitswerte von Formeln unter Belegungen
- ▶ Modelle von Formeln und Formelmengen
- ▶ Modellmenge und WW-Tabelle von Formeln und Formelmengen
- ▶ erfüllbare, allgemeingültige Formeln
- ▶ semantische Äquivalenz von Formeln

Normalformen aussagenlogischer Formeln

spezielle aussagenlogische Formeln:

Atom Aussagenvariable

Literal Atom oder negiertes Atom

Klausel Disjunktion von Literalen

Normalformen:

NNF Formeln, in denen das Negationssymbol $\neg$ höchstens auf Atome angewendet wird, heißen in **Negations-Normalform**.

Beispiel: $\neg p \vee ((\neg q \vee p) \wedge q)$, $\neg p$, p

CNF Formeln der Form $\bigwedge_{i=1}^n \left(\bigvee_{j=1}^{m_i} l_{i,j} \right)$ mit Literalen $l_{i,j}$ heißen in **konjunktiver Normalform**. (Konjunktion von Klauseln)

Beispiel: $(\neg p \vee \neg q) \wedge (p \vee q) \wedge \neg q$, $p \vee q$, $p \wedge \neg q$, $\neg p$

DNF Formeln der Form $\bigvee_{i=1}^n \left(\bigwedge_{j=1}^{m_i} l_{i,j} \right)$ mit Literalen $l_{i,j}$ heißen in **disjunktiver Normalform**.

Beispiel: $\neg p \vee (\neg q \wedge p) \vee (p \wedge q)$, $p \vee q$, $p \wedge \neg q$, $\neg p$

Satz über Normalformen

Satz

Zu jeder Formel $\varphi \in \text{AL}(P)$ existieren

- ▶ eine äquivalente Formel $\psi \in \text{AL}(P)$ in NNF,*
- ▶ eine äquivalente Formel $\psi' \in \text{AL}(P)$ in CNF und*
- ▶ eine äquivalente Formel $\psi'' \in \text{AL}(P)$ in DNF.*

Transformation beliebiger Formeln in Normalformen:

1. $\{\vee, \wedge, \neg\}$ ist eine Junktorbasis, d.h.
Formeln mit Junktoren $\rightarrow, \leftrightarrow, \uparrow, \Downarrow$ lassen sich in äquivalente Formeln mit ausschließlich den Junktoren $\vee, \wedge, \neg$ umformen
2. Konstruktion einer NNF durch (ggf. mehrmalige) Anwendung der deMorganschen Regeln
3. Konstruktion der CNF und DNF durch (ggf. mehrmalige) Anwendung der Distributivgesetze auf die NNF

Normalformen – Beispiel

$$\begin{aligned}(a \rightarrow b) \rightarrow c & \stackrel{(\varphi \rightarrow \psi \equiv \neg \varphi \vee \psi)}{=} (\neg a \vee b) \rightarrow c \\ & \stackrel{(\varphi \rightarrow \psi \equiv \neg \varphi \vee \psi)}{=} \neg(\neg a \vee b) \vee c \\ & \stackrel{(\text{deMorgan})}{=} (\neg \neg a \wedge \neg b) \vee c \\ & \stackrel{(\neg \neg \varphi \equiv \varphi)}{=} (a \wedge \neg b) \vee c && (\text{NNF, DNF}) \\ & \stackrel{(\text{Distributivität})}{=} (a \vee c) \wedge (\neg b \vee c) && (\text{NNF, CNF})\end{aligned}$$

Modellierungsbeispiel Taxi

umgangssprachliche Angaben zur **Situation**:

Wenn der Zug zu spät kommt und kein Taxi am Bahnhof steht, ist Tom nicht pünktlich. Der Zug kam zu spät und Tom ist pünktlich.

Frage: Stand ein Taxi am Bahnhof?

formale Beschreibung:

- ▶ elementare Aussagen (Atome): $P = \{p, t, z\}$
 p – Tom ist pünktlich, t – Taxi steht da, z – Zug hat Verspätung
- ▶ Beschreibung der Situation (Kontext) als **Formelmenge**:

$$\Phi = \{(z \wedge \neg t) \rightarrow \neg p, z \wedge p\}$$

- ▶ Modellierung der Frage als **Behauptung** : t
(positive Antwort auf die Frage)

Alle möglichen Situationen, in denen alle Angaben erfüllt sind:

$\text{Mod}(\Phi) = \dots$

Antwort auf die Frage: ja,

weil für jede Belegung $W \in \text{Mod}(\Phi)$ gilt $W(t) = 1$,

also **$\text{Mod}(\Phi) \subseteq \text{Mod}(t)$**

Semantisches Folgern

Definition

Für jede Formelmenge $\Phi \subseteq \text{AL}(P)$ (Kontext)
und jede Formel $\psi \in \text{AL}(P)$ (Behauptung) heißt
 ψ genau dann **semantische Folgerung** aus Φ wenn $\text{Mod}(\Phi) \subseteq \text{Mod}(\psi)$.

Notation: $\Phi \models \psi$ (ψ folgt semantisch aus Φ)

Beispiele:

- ▶ $\{p, p \rightarrow q\} \models q$ gilt, weil
 $\text{Mod}(\{p, p \rightarrow q\}) = \{W_{11}\} \subseteq \{W_{01}, W_{11}\} = \text{Mod}(q)$,
- ▶ $\{p, \neg(q \wedge p)\} \models \neg q$, weil ...
- ▶ $p \models q \rightarrow p$, weil ...
- ▶ $q \rightarrow p \not\models p$, weil ...
- ▶ $\emptyset \models p \vee \neg p$, weil ...
- ▶ $\emptyset \not\models p$, weil ...

Notation für Spezialfälle

für $\Phi = \{\varphi\}$	:	φ	$\models$	ψ	(statt	$\{\varphi\}$	$\models$	ψ)
für $\Phi = \emptyset$	:		$\models$	ψ	(statt	$\emptyset$	$\models$	ψ)

Sätze über das Folgern

Satz

Für jede endliche Formelmeng $\Phi = \{\varphi_1, \dots, \varphi_n\} \subseteq \text{AL}(P)$ gilt

$$\Phi \models \psi \quad \text{genau dann, wenn} \quad \bigwedge_{i=1}^n \varphi_i \models \psi$$

Beweis: verwendet Fakt auf Folie 46

Satz

Für jede Formelmeng $\Phi \subseteq \text{AL}(P)$ und jede Formel $\psi \in \text{AL}(P)$ gilt

$\Phi \models \psi$ genau dann, wenn $\Phi \cup \{\neg\psi\}$ **unerfüllbar** ist.

Beweis

Satz

Für jede Formel $\psi \in \text{AL}(P)$ gilt $\models \psi$ genau dann, wenn ψ allgemeingültig.

Satz

Für je zwei beliebige Formeln $\varphi, \psi \in \text{AL}(P)$ gilt $\varphi \equiv \psi$ genau dann, wenn

1. $\varphi \models \psi$ und $\psi \models \varphi$ gilt,
2. die Formel $\varphi \leftrightarrow \psi$ allgemeingültig ist.

Was bisher geschah

Modellierung von **Aussagen** in (klassischer) Aussagenlogik

Syntax:

- ▶ induktive Definitionen von Formelmengen: $AL(P)$, $AL_J(P)$
- ▶ Baumstruktur von Formeln
- ▶ Beweise durch strukturelle Induktion
- ▶ Junktorbasen
- ▶ Normalformen: NNF, CNF, DNF

Semantik:

- ▶ Wahrheitswerte von Formeln unter Belegungen
- ▶ Modelle von Formeln und Formelmengen
- ▶ Modellmengen von Formeln und Formelmengen
- ▶ erfüllbare, allgemeingültige Formeln
- ▶ semantische Äquivalenz von Formeln
- ▶ semantisches Folgern von Formeln aus Formelmengen
- ▶ Transformation von Folgerungs- in Erfüllbarkeitsprobleme

Modellierungsbeispiel: Winterkleidung

- ▶ Wenn es kalt ist, trägt Paul immer eine Mütze, einen Schal oder Handschuhe.
- ▶ Ohne Handschuhe oder Schal trägt er keine Mütze.
- ▶ Mütze und Handschuhe trägt er nie zusammen.
- ▶ Handschuhe und Schal trägt er immer zugleich.

Modellierung als (aussagenlogische) Formelmenge

$$\Phi = \{m \vee s \vee h, \neg(h \vee s) \rightarrow \neg m, \neg(m \wedge h), h \leftrightarrow s\}$$

Welche der folgenden Aussagen sind semantische Folgerungen daraus:

- ▶ Bei Kälte trägt er immer einen Schal. $\psi_1 = s$
- ▶ Bei Kälte trägt er immer Handschuhe oder Mütze. $\psi_2 = h \vee m$
- ▶ Er trägt nie Handschuhe. $\psi_3 = \neg h$

schon bekannt: Lösung durch

- ▶ Modellmengen $\text{Mod}(\Phi) \subseteq \text{Mod}(\psi)$

bei Problemen mit vielen Variablen aufwendig

gut wäre: **syntaktisches** Verfahren

Typische praktische Folgerungsprobleme

- ▶ kombinatorische Probleme, z.B. Planungsprobleme (Registerzuweisung, Zeitplanung), Graphfärbung
- ▶ Verifikation digitaler Schaltungen:

gegeben: Schaltung:

(Ausgabe-Verhalten als boolesche Funktion, Formel φ)
Spezifikation (logische Formel ψ)

Frage: Erfüllt die Schaltung die Spezifikation? (Gilt $\varphi \models \psi$?)

Gegenbeispiel bei negativer Antwort:

Belegung $W : P \rightarrow \{0, 1\}$ mit $W(\varphi \wedge \neg\psi) = 1$

- ▶ Verifikation von Software:

gegeben: Programm (Ausgabe-Verhalten als Formelmenge Φ)
Spezifikation (logische Formel ψ)

Frage: Erfüllt das Programm die Spezifikation? (Gilt $\Phi \models \psi$?)

Gegenbeispiel bei negativer Antwort:

Belegung $W : P \rightarrow \{0, 1\}$ mit $W \in \text{Mod}(\Phi \cup \{\neg\psi\})$

Syntaktisches Ableiten – Ziel

gegeben: Formelmenge $\Phi \subseteq \text{AL}(P)$
und Formel $\psi \in \text{AL}(P)$

Frage : Gilt $\Phi \models \psi$?

Ziel:

Verfahren zur Beantwortung dieser Frage durch **syntaktische**
Operationen (ohne Verwendung der Semantik, Modellmengen)

Logische Kalküle

Verfahren zur **schrittweisen** syntaktischen Ableitung
von semantischen Folgerungen (Formeln)
aus einer Menge von Formeln (Hypothesen)

Definition von

Axiomen (Schemata für) Formeln

Schlussregeln zur syntaktischen Ableitung von Folgerungen aus
Formelmengen (Axiomen, Hypothesen, Annahmen)
oft mehrere Voraussetzungen und eine Folgerung

Ableitungen (und Beweisen) durch geeignete Kombinationen von
Schlussregeln (Baumstruktur)

Kalküle - Definition

Kalkül $K = (A, R)$ mit Mengen von

Axiome A , d.h. (Schemata für) Formeln, deren Gültigkeit vorausgesetzt wird,

Schlussregeln R zur **syntaktischen Ableitung** von **Folgerungen** (weitere gültige Formeln) aus Formelmengen (Axiomen und Hypothesen)

häufige Darstellung von Schlussregeln als Verzweigung (nach oben):

$$\frac{\varphi_1 \quad \cdots \quad \varphi_n}{\psi}$$

- ▶ Voraussetzungen (oben): $\varphi_1, \dots, \varphi_n$
- ▶ Folgerung (unten): ψ

Beweis im Kalkül = Baum mit

- ▶ Blätter: Axiome
- ▶ innere Knoten: (instanziierte) Schlussregeln

Aussagenlogische Resolution – Idee

Folgerungsproblem: gegeben: Formelmenge Φ
und Formel ψ

Frage: Gilt $\Phi \models \psi$?

Idee:

schrittweise Erweiterung der Formelmenge Φ durch
Hinzufügen von semantischen Folgerungen ψ aus Φ
(also Formeln ψ mit $\Phi \models \psi$).

Dabei bleibt die Modellmenge unverändert,
d.h. $\text{Mod}(\Phi \cup \{\psi\}) = \text{Mod}(\Phi)$.

(Warum?)

Spezialfall $\psi = \text{f}$:

$\Phi \models \text{f}$ gilt genau dann, wenn Φ unerfüllbar ist
(einen Widerspruch enthält).

Aussagenlogischer Resolutionskalkül

Resolutionsregel für $\psi, \eta \in \text{AL}(P)$ und **Aussagenvariable** $p \in P$:

$$\frac{\psi \vee p \quad \neg p \vee \eta}{\psi \vee \eta} \text{ (R)}$$

alternative Darstellung:

$$\frac{\neg\psi \rightarrow p \quad p \rightarrow \eta}{\neg\psi \rightarrow \eta} \text{ (R)}$$

gegeben: Formelmenge Φ

Formel ψ

Frage : Gilt $\Phi \models \psi$?

Idee: Schrittweise Erweiterung der Formelmenge Φ um semantische Folgerungen der Form $\psi \vee \eta$, falls $\{p \vee \psi, \neg p \vee \eta\} \subseteq \Phi$
(Anwendung der Resolutionsregel)

Spezialfall: Φ enthält nur Klauseln

Mengendarstellung von Formeln in CNF

$$\text{CNF} \quad \varphi = \bigwedge_{i=1}^n \bigvee_{j=1}^{m_i} l_{i,j}$$

WH: Disjunktionen $\bigvee_{j=1}^{m_i} l_{i,j}$ heißen **Klauseln**

Spezialfall: **leere Klausel** mit $m_i = 0$

$$\bigvee_{i=1}^0 l_i = \bigvee_{i \in \emptyset} l_i \equiv \text{f}$$

Repräsentation der CNF φ als Menge von Klauseln:

$$\varphi = \bigwedge_{i=1}^n \bigvee_{j=1}^{m_i} l_{i,j} \quad \mapsto \quad \left\{ \bigvee_{j=1}^{m_i} l_{i,j} \mid i \in \{1, \dots, n\} \right\}$$

Beispiele für CNF als Klauselmengen

Notation:

- ▶ Jede Klausel enthält jedes Literal nur einmal.
(keine Einschränkung, weil $\forall \varphi \in AL(P) : \varphi \vee \varphi \equiv \varphi$)
- ▶ Literale in jeder Klausel in alphabetischer Reihenfolge
(keine Einschränkung weil $\vee$ kommutativ).
- ▶ Mengen enthalten jede Klausel nur einmal.
(keine Einschränkung, weil $\forall \varphi \in AL(P) : \varphi \wedge \varphi \equiv \varphi$)

$$a \vee b \vee \neg c \vee a \mapsto \{a \vee b \vee \neg c\}$$

$$a \wedge b \wedge \neg c \wedge a \mapsto \{a, b, \neg c\}$$

$$(a \vee \neg b \vee a) \wedge (b \vee a) \wedge (\neg c \vee a) \mapsto \{a \vee \neg b, a \vee b, a \vee \neg c\}$$

$$(a \vee \neg b) \wedge (\neg c \vee a) \wedge (a \vee b) \wedge (b \vee a) \mapsto \{a \vee \neg b, a \vee b, a \vee \neg c\}$$

Resolutionsprinzip

Regel allgemein:

$$\frac{\psi \vee \textcolor{red}{l} \quad \eta \vee \neg \textcolor{red}{l}}{\psi \vee \eta} \text{ (R)}$$

Regel für Klauseln:

$$\frac{l_1 \vee \dots \vee l_n \vee \textcolor{red}{l} \quad l'_1 \vee \dots \vee l'_m \vee \neg \textcolor{red}{l}}{l'_1 \vee \dots \vee l'_m} \text{ (R)}$$

Resolvente $R = l_1 \vee \dots \vee l_n \vee l'_1 \vee \dots \vee l'_m$

Beispiele:

- ▶ Klauseln $K_1 = a \vee \neg b \vee \textcolor{red}{c}$ und $K_2 = \neg \textcolor{red}{c} \vee d \vee \neg e$
haben eine Resolvente $R = a \vee \neg b \vee d \vee \neg e$
- ▶ Klauseln $K_1 = a \vee \neg b$ und $K_2 = \neg a \vee b \vee \neg c$
haben zwei Resolventen $R_1 = \neg b \vee b \vee \neg c$, $R_2 = \neg a \vee a \vee \neg c$

Satz (Resolutionslemma)

Für jede CNF (Klauselmenge) Φ und die Resolvente R zweier Klauseln aus Φ gilt

$$\text{Mod}(\Phi) = \text{Mod}(\Phi \cup \{R\})$$

Ableitungen durch Resolution

Resolutionsableitung aus einer Klauselmenge Φ (CNF):
endliche Folge $C_1, \dots, C_n$ von Klauseln, wobei für jede Klausel C_i gilt:

- ▶ $C_i \in \Phi$ oder
- ▶ C_i ist eine Resolvente von Klauseln C_j, C_k mit $j < i$ und $k < i$.

Resolutionsableitung **der Klausel ψ** aus Klauselmenge Φ :

Resolutionsableitung $C_1, \dots, C_n$ in Φ mit **$C_n = \psi$**

Beispiel: Resolutionsableitung von d aus

$$\Phi = \{\underbrace{a \vee b \vee c}_{C_1}, \underbrace{\neg b \vee d}_{C_2}, \underbrace{\neg a \vee d}_{C_3}, \underbrace{\neg c \vee d}_{C_4}\}$$

$$R(C_1, C_4) = \underbrace{a \vee b \vee d}_{C_5}, \quad R(C_3, C_5) = \underbrace{b \vee d}_{C_6}, \quad R(C_2, C_6) = d$$

Baumdarstellung

Resolutionsableitungen nach $\mathbb{f}$

Problem:

Es existiert **keine** Resolutionsableitung von $\neg a \vee \neg b \vee d$ aus

$$\Phi = \{a \vee b \vee c, \neg b \vee d, \neg a \vee d, \neg c \vee d\}$$

aber es gilt $\Phi \models \neg a \vee \neg b \vee d$.

Lösung:

- ▶ Es gilt $\Phi \models \psi$ gdw. $\Phi \cup \{\neg\psi\}$ unerfüllbar.
- ▶ Unerfüllbarkeitsbeweis für $\Phi \cup \{\neg\psi\}$ durch Resolutionsableitung von $\mathbb{f}$ aus $\Phi \cup \{\neg\psi\}$ (Klauselform)

Beispiel: Resolutionsableitung von $\mathbb{f}$ aus

$$\Phi \cup \{\neg\psi\} = \{a \vee b \vee c, \neg b \vee d, \neg a \vee d, \neg c \vee d, a, b, \neg d\}$$

Was bisher geschah

Modellierung von **Aussagen** in (klassischer) Aussagenlogik

Syntax:

- ▶ induktive Definitionen von Formelmengen: $AL(P)$, $AL_J(P)$
- ▶ Baumstruktur von Formeln
- ▶ Beweise durch strukturelle Induktion
- ▶ Junktorbasen
- ▶ Normalformen: NNF, CNF, DNF
- ▶ Aussagenlogischer Resolutionskalkül

Semantik:

- ▶ Wahrheitswerte von Formeln unter Belegungen
- ▶ Modelle von Formeln und Formelmengen
- ▶ Modellmengen von Formeln und Formelmengen
- ▶ erfüllbare, allgemeingültige Formeln
- ▶ semantische Äquivalenz von Formeln
- ▶ semantisches Folgern von Formeln aus Formelmengen
- ▶ Transformation von Folgerungs- in Erfüllbarkeitsprobleme

Ableitungen im Resolutionskalkül

Schon gezeigt:

Für jede Formelmengende $\Phi \subseteq \text{AL}(P)$ und jede Formel $\psi \in \text{AL}(P)$ gilt:

$$\Phi \models \psi \quad \text{gdw.} \quad \Phi \cup \{\neg\psi\} \text{ unerfüllbar}$$

Die Formel ψ ist genau dann aus der Formelmengende Φ **durch aussagenlogische Resolution ableitbar**, wenn eine Resolutionsableitung für $\mathbb{f}$ aus $\Phi \cup \{\neg\psi\}$ existiert.

Beispiele:

- ▶ e ist durch aussagenlogische Resolution aus $\{a \vee b \vee c, (a \vee b) \rightarrow d, c \rightarrow e, \neg d\}$ ableitbar.
- ▶ $(\neg p \vee q) \wedge (\neg q \vee r) \wedge p \wedge \neg r$ ist unerfüllbar.
- ▶ $\varphi = (q \wedge r) \vee (\neg p \wedge \neg q \wedge r) \vee p \vee (\neg p \wedge \neg r)$ ist allgemeingültig.

Modellierungsbeispiel Taxi (Resolution)

Wenn der Zug zu spät kommt und kein Taxi am Bahnhof steht, ist Tom nicht pünktlich. Der Zug kam zu spät und Tom ist pünktlich.

$$\Phi = \{(z \wedge \neg t) \rightarrow \neg p, z \wedge p\}$$

Frage: Stand ein Taxi am Bahnhof? $\psi = t$

Ableitung von ψ aus Φ durch Resolution:

1. Transformation von Φ in Klauselmenge: $\left\{ \underbrace{\neg z \vee t \vee \neg p}_1, \underbrace{z}_2, \underbrace{p}_3 \right\}$
2. direkte Resolutionsableitung von t aus Φ (funktioniert hier auch)

$$R(3, 1, p) = \neg z \vee t \quad (4)$$

$$R(2, 4, z) = t \quad (5)$$

- 2'. Resolutionsableitung von ff aus $\Phi \cup \{\neg t\}$ (4) (funktioniert immer)

$$R(3, 1, p) = \neg z \vee t \quad (5)$$

$$R(2, 5, z) = t \quad (6)$$

$$R(6, 4, t) = \text{ff}$$

WH: Normalformen

CNF: $\bigwedge_{i=1}^n \left(\bigvee_{j=1}^{m_i} l_{i,j} \right)$ DNF: $\bigvee_{i=1}^n \left(\bigwedge_{j=1}^{m_i} l_{i,j} \right)$ mit Literalen $l_{i,j}$

Satz

Zu jeder Formel $\varphi \in \text{AL}(P)$ existieren

- ▶ *eine äquivalente Formel $\psi \in \text{AL}(P)$ in NNF,*
- ▶ *eine äquivalente Formel $\psi' \in \text{AL}(P)$ in CNF und*
- ▶ *eine äquivalente Formel $\psi'' \in \text{AL}(P)$ in DNF.*

Transformation beliebiger Formeln in Normalformen:

1. $\{\vee, \wedge, \neg\}$ ist eine Junktorbasis, d.h.
Formeln mit Junktoren $\rightarrow, \leftrightarrow, \text{t}, \text{f}$ lassen sich in äquivalente Formeln mit ausschließlich den Junktoren $\vee, \wedge, \neg$ umformen
2. Konstruktion einer NNF durch (ggf. mehrmalige) Anwendung der deMorganschen Regeln
3. Konstruktion der CNF und DNF durch (ggf. mehrmalige) Anwendung der Distributivgesetze auf die NNF

Erfüllbarkeitsproblem für AL (SAT)

Aufgabe **SAT**:

gegeben: **beliebige Formel** $\varphi \in \text{AL}(P)$

Frage: Ist φ **erfüllbar** (satisfiable) ? (Gilt $\text{Mod}(\varphi) \neq \emptyset$?)

Instanz, z.B.

$$\varphi = ((z \wedge \neg t) \rightarrow \neg p) \wedge (z \wedge p) \wedge \neg t$$

einige mögliche Lösungsverfahren:

semantisch : Berechnung der Modellmenge (WW-Tabelle)

syntaktisch :

- ▶ äquivalente Umformungen ($\varphi \equiv \mathbb{f}$)
- ▶ aussagenlogische Resolution von $\mathbb{f}$ aus φ

DNF-SAT

Aufgabe **DNF-SAT**: gegeben: **DNF** $\varphi = \bigvee_{i=1}^m \bigwedge_{j=1}^{k_i} l_{i,j} \in \text{AL}(P)$

Frage: Ist φ **erfüllbar**?

Instanz, z.B. $(p \wedge \neg q \wedge \neg p) \vee (q \wedge p \wedge \neg q) \vee (\neg p \wedge \neg q)$

Lösungsidee:

- ▶ φ ist genau dann erfüllbar, wenn (wenigstens) eine der m Konjunktionen $\bigwedge_{j=1}^{k_i} l_{i,j}$ erfüllbar ist.
- ▶ Konjunktion $\bigwedge_{j=1}^{k_i} l_{i,j}$ ist genau dann unerfüllbar, wenn für eine Aussagenvariable $x \in \text{var}(\varphi)$ gilt: $\{x, \neg x\} \subseteq \{l_{i,j} \mid j \in \{1, \dots, k_i\}\}$ (Widerspruch).

Lösungsverfahren:

- ▶ für jede Konjunktion der DNF Widerspruchsfreiheit überprüfen,
Falls eine widerspruchsfreie Konjunktion gefunden:
DNF erfüllbar, sonst DNF unerfüllbar
- ▶ Überprüfung der Widerspruchsfreiheit einer Konjunktion:
für jedes Literal überprüfen, ob Negation in derselben Konjunktion,
falls gefunden: diese Konjunktion unerfüllbar

DNF-SAT ist einfach (schnell) zu lösen.

CNF-SAT

Aufgabe **CNF-SAT**:

gegeben: **CNF** $\varphi = \bigwedge_{i=1}^m \bigvee_{j=1}^{k_i} l_{i,j} \in \text{AL}(P)$

Frage: Ist φ **erfüllbar**?

Instanz z.B. $(p \vee \neg q \vee r) \wedge (q \vee p \vee \neg r) \wedge (\neg p \vee \neg q)$

Lösungsansätze:

- ▶ Test aller möglichen Belegungen, hoher Aufwand
(für große Anzahl an Aussagenvariablen unpraktikabel)
- ▶ Umformung in eine zu φ äquivalente DNF ψ
Test von ψ auf Erfüllbarkeit
(für große Anzahl an Aussagenvariablen unpraktikabel)
- ▶ Konstruktion einer Formel ψ mit
 1. ψ erfüllbar gdw. φ erfüllbar und
 2. Erfüllbarkeit für ψ einfach zu testen

CNF-SAT ist die dem menschlichen Formulieren von Bedingungen naheliegendste Form, aber **schwierig zu lösen**. (hoher Zeitaufwand)

2-CNF und 2SAT

2-CNF: Formel der Form

$$\varphi = \bigwedge_{i=1}^m \bigvee_{j=1}^{k_i} l_{i,j} \quad \text{mit} \quad \forall i \in \{1, \dots, m\} : k_i \leq 2$$

Beispiele:

- ▶ $(p \vee q) \wedge (\neg p \vee r) \wedge \neg q$ ist 2-CNF,
- ▶ $(p \vee q) \wedge (\neg p \vee q) \wedge (p \vee \neg q) \wedge (\neg p \vee \neg q)$ ist 2-CNF,
- ▶ $q \vee \neg p \vee r$ ist CNF (und DNF), aber keine 2-CNF

Aufgabe **2SAT**:

gegeben: **2-CNF** φ

Frage: Ist φ **erfüllbar**?

2SAT ist einfach (schnell) zu lösen.

(hier ohne Beweis)

Aber:

Nicht zu jeder Formel $\varphi \in \text{AL}(P)$ existiert eine äquivalente 2-CNF.

3-CNF und 3SAT

3-CNF: Formel der Form

$$\varphi = \bigwedge_{i=1}^m \bigvee_{j=1}^{k_i} l_{i,j} \quad \text{mit} \quad \forall i \in \{1, \dots, m\} : k_i \leq 3$$

Instanz z.B. $(p \vee \neg q \vee r) \wedge (q \vee p \vee \neg r) \wedge (\neg p \vee \neg q)$

Aufgabe **3SAT**:

gegeben: **3-CNF** φ

Frage: Ist φ **erfüllbar**?

bekannt (hier ohne Beweis):

Zu jeder Formel $\varphi \in \text{AL}(P)$ lässt sich eine erfüllbarkeitsäquivalente 3-CNF konstruieren.

bekannt (Beweis im Master-Studium):

3SAT ist schwierig zu lösen. (zeitaufwendig, NP-vollständig)

SAT-Solver

SAT-Solver: Werkzeug zum Lösen von (CNF-)SAT-Instanzen

SAT-Solver

- ▶ benutzen heuristische Verfahren,
- ▶ finden für praktische Probleme oft schnell eine Lösung,
- ▶ meist Ausgabe eines Modells (wenn wenigstens eins existiert)

aktive Forschung auf diesem Gebiet:

jährlich Wettbewerbe (www.satcompetition.org/)

typische Anwendung von SAT-Solvern:

1. Modellierung des ursprünglichen Problems P als CNF φ in DIMACS-Format (oft maschinell),
2. Lösung mit SAT-Solver
3. Übersetzung erfüllender Belegung für φ in Lösung für P

Modellierungsbeispiel: Zuordnung Person – Land

In einem Eisenbahnabteil sitzen die Herren Lehmann und Müller.
Einer ist Sachse und einer Thüringer.

Welche Zuordnungen sind möglich?

Modellierung in Aussagenlogik:

Aussagenvariablen: $P = \{LS, LT, MS, MT\}$

LS Herr Lehmann ist Sachse.

LT Herr Lehmann ist Thüringer.

MS Herr Müller ist Sachse.

MT Herr Müller ist Thüringer.

Jeder der beiden Personen kommt aus genau einem Land, also

1. Jede Person kommt aus wenigstens einem Land.

$$LS \vee LT, MS \vee MT$$

2. Aus jedem Land kommt wenigstens eine Person.

$$LS \vee MS, LT \vee MT$$

3. Jede Person kommt aus höchstens einem Land.

$$LS \rightarrow \neg LT, LT \rightarrow \neg LS, MS \rightarrow \neg MT, MT \rightarrow \neg MS$$

oder (äquivalent) $\neg LS \vee \neg LT, \neg MS \vee \neg MT$

4. Aus jedem Land kommt höchstens eine Person. (analog)

Lösung mit SAT-Solver

Eingabe im DIMACS-Format für CNF (ASCII):

- ▶ erste Zeile enthält Typ (cnf), Anzahl der Aussagenvariablen und Disjunktionen (z.B. p cnf 4 8)
- ▶ Aussagenvariablen $\{1, \dots, n\}$
- ▶ jede Disjunktion (Klausel) eine Zeile, 0 markiert Ende der Klausel
– statt $\neg$, Literale durch Leerzeichen getrennt,

Darstellung der Zuordnungs-Aufgabe als CNF in DIMACS-Format

```
p cnf 4 8
c 1:LS, 2:LT, 3:MS, 4:MT
1 2 0
3 4 0
...
```

Lösung mit SAT-Solver, z.B. minisat, Ausgabe: eine erfüllende Belegung

SATISFIABLE

```
1 -2 -3 4 0
```

Interpretation der SAT-Solver-Ausgabe als (ein) Modell:

LS (1) und MT (4) sind wahr, also Lehmann Sachse, Müller Thüringer.
für weitere Modelle zusätzliche Klauseln, die bekanntes Modell verhindern
im Beispiel: Zeile -1 2 3 -4 0 hinzufügen

Einsatz von SAT-Solvern

typische Anwendungen für SAT-Solver z.B.

- ▶ Zuordnungen (mit vielen beteiligten Individuen), z.B. Ressourcen-Planung
 - ▶ Aufgabenverteilung
 - ▶ Planen, z.B. Stundenplan
 - ▶ Job-Scheduling (Betriebssystem)
- ▶ Constraint-Lösen, kombinatorische Suchprobleme, z.B. Graph-Färbungen (Register-Zuordnung, Sudoku)
- ▶ Folgerungsprobleme, z.B.
 - ▶ Schaltkreisentwurf und -verifikation
 - ▶ Konfiguration von Hard- und Software
 - ▶ Model-Checking (Verifikation von Software und Systemen)

Was bisher geschah

Modellierung von **Aussagen**
zur Beschreibung von Situationen, Anforderungen
(klassische) **Aussagenlogik**

- ▶ Syntax: Atome sind Aussagenvariablen, Junktoren
 $\neg, \vee, \wedge, \rightarrow, \leftrightarrow$
induktive Definition: Baumstruktur der Formeln
strukturelle Induktion
- ▶ Semantik: Belegungen, WW-Tabellen, Modellmengen
- ▶ erfüllbare, allgemeingültige Formeln, Äquivalenz von Formeln
- ▶ semantisches Folgern, syntaktisches Schließen
(Resolutionskalkül)

Erfüllbarkeits-Aufgaben

- ▶ Modellierungsbeispiele, Anwendungen
- ▶ SAT-Solver
- ▶ Transformation von Allgemeingültigkeits- und Folgerungsfragen in Erfüllbarkeitsfragen

Modellierung von Daten durch Mengen

Naiver Mengenbegriff von Georg Cantor (1845-1918):

Eine Menge ist eine Zusammenfassung bestimmter, wohlunterschiedener Dinge unserer Anschauung oder unseres Denkens, welche Elemente der Menge genannt werden, zu einem Ganzen.

Mengen werden dargestellt:

extensional durch Angabe aller Elemente

(nur für endliche Mengen möglich)

Beispiel: $\{0, 1, 2, 3\}$, $\{\{a\}, 5, \{a, b\}\}$

intensional durch Angabe der gemeinsamen Eigenschaft aller Elemente (oft durch prädikatenlogische Formeln)

Beispiele:

$$\{x \mid (x \in \mathbb{N}) \wedge (x < 4)\} = \{x \in \mathbb{N} \mid x < 4\}$$

$$\{x \mid x \in \mathbb{N} \wedge \exists y((y \in \mathbb{N}) \wedge (x = 2y + 1))\}$$

Modellierungsbeispiel Skatkarten (extensional)

Modellierung:

- ▶ Jede Karte hat Farbe und Wert (wichtig)
- ▶ deutsches oder französisches Blatt? (egal)
- ▶ Ordnung zwischen Karten derselben Farbe (wichtig)
z.B. $7 < 8 < 9 < 10 < B < D < K < A$ bei Null-Spiel
- ▶ Bedienregeln
- ▶ Trumpfregeln

Formale Darstellung der Karten:

Jede Karte ist eindeutig repräsentiert durch Paar von

Farbe aus der Menge $F = \{\diamondsuit, \heartsuit, \spadesuit, \clubsuit\}$ und

- Wert**
- ▶ Zahl aus der Menge $Z = \{7, 8, 9, 10\}$ oder
 - ▶ Bild aus der Menge $B = \{B, D, K, A\}$

Modellierungsbeispiel gerade Zahlen (intensional)

Gerade Zahlen sind genau die durch 2 teilbaren ganzen Zahlen.

Individuen: $\mathbb{Z}$ (Menge aller ganzen Zahlen)

Eigenschaften: gerade

Beziehungen: Teilbarkeit ganzer Zahlen ($a|b$)

präziser mit Definition der Teilbarkeit:

$x \in \mathbb{Z}$ ist genau dann gerade, wenn $2|x$

$x \in \mathbb{Z}$ ist genau dann gerade, wenn ein $y \in \mathbb{Z}$ mit $2y = x$ existiert.

Bedingung für „ x ist eine gerade Zahl“ als prädikatenlogische Formel:

$$(x \in \mathbb{Z}) \wedge \exists y((y \in \mathbb{Z}) \wedge (2y = x))$$

Menge aller geraden Zahlen:

$$\begin{aligned} & \{x \mid (x \in \mathbb{Z}) \wedge \exists y((y \in \mathbb{Z}) \wedge (2y = x))\} \\ &= \{x \in \mathbb{Z} \mid \exists y((y \in \mathbb{Z}) \wedge (2y = x))\} \\ &= \{2y \mid y \in \mathbb{Z}\} = 2\mathbb{Z} \end{aligned}$$

Was sind analog $3\mathbb{Z}$, $1\mathbb{Z}$, $0\mathbb{Z}$?

Endliche Mengen

Menge A heißt **endlich** gdw. eine Zahl $n \in \mathbb{N}$ existiert, so dass A genau n Elemente enthält.

Die leere Menge $\emptyset$ ist die (eindeutig bestimmte) Menge, die kein Element enthält.

Beispiele:

- ▶ $\{x \mid (x \in \mathbb{N}) \wedge (x \leq 10)\}$ endlich
- ▶ $\{x \mid (x \in \mathbb{N}) \wedge (x \text{ ist gerade}) \wedge (x < 100)\}$ endlich
- ▶ $\{x \mid (x \in \mathbb{N}) \wedge (x \text{ ist gerade}) \wedge (x > 100)\}$ nicht endlich
- ▶ $\{x \mid (x \in \mathbb{Z}) \wedge (x \text{ ist gerade}) \wedge (x < 100)\}$ nicht endlich
- ▶ $\{x \mid (x \in \mathbb{Z}) \wedge (x \text{ ist gerade}) \wedge (x \text{ ist ungerade})\} = \emptyset$ endlich
- ▶ $\{x \mid x \text{ ist Primzahl}\}$ nicht endlich (Widerspruchsbeweis)

$|A|$ heißt **Mächtigkeit (Kardinalität)** von A .

Beispiele:

- ▶ $|\{\diamond, \heartsuit, \spadesuit, \clubsuit\}| = 4$
- ▶ $|\{\text{rot}, \text{grün}, \text{blau}\}| = 3$
- ▶ $|\{x \mid x \in \mathbb{N} \wedge x \leq 10\}| = 11$
- ▶ $|\{a, b, \{a, b\}, \{a, a, b\}, b\}| = 3$
- ▶ $|\emptyset| = 0$

Beziehungen zwischen Mengen

∈ Element-Relation

Notation $x \in M$ für: x ist **Element** der Menge M

Negation kurz $\notin$: $a \notin M$ gdw. $\neg(a \in M)$

Beispiele: $a \in \{a, b, 2\}$, $\{a\} \notin \{a, b, 2\}$

⊆ Teilmengen-Relation

$A \subseteq B$ gdw. $\forall x (x \in A \rightarrow x \in B)$

Beispiele: $\{a, b\} \subseteq \{2, a, b, \{2\}\}$, $\{a\} \subseteq \{a\}$,
 $\{a\} \not\subseteq \{\{a\}\}$, aber $\{a\} \in \{\{a\}\}$

= Mengengleichheit

$A = B$ gdw. $\forall x (x \in A \leftrightarrow x \in B)$

Beispiele: $\{a, b, 2\} = \{2, a, b, 2\}$, $\{a, b, 2\} \neq \{a, \{b\}, 2\}$

⊂ echte Teilmenge

$A \subset B$ gdw. $(A \subseteq B) \wedge \neg(A = B)$

Beispiele: $\{2, a\} \subset \{a, b, 2\}$,
 $\{2, a\} \not\subset \{2, a\}$, $\{2, a, b\} \not\subset \{2, a\}$

Operationen auf Mengen

Vereinigung $A \cup B = \{x \mid (x \in A) \vee (x \in B)\}$
 $(x \in (A \cup B)) \leftrightarrow ((x \in A) \vee (x \in B))$
Beispiel: $\{a, c, d\} \cup \{b, c\} = \{a, b, c, d\}$

Schnitt $A \cap B = \{x \mid (x \in A) \wedge (x \in B)\}$
 $(x \in (A \cap B)) \leftrightarrow ((x \in A) \wedge (x \in B))$
Beispiel: $\{a, c, d\} \cap \{b, c\} = \{c\}$

Differenz $A \setminus B = \{x \mid (x \in A) \wedge \neg(x \in B)\}$
 $(x \in (A \setminus B)) \leftrightarrow ((x \in A) \wedge \neg(x \in B))$
Beispiel: $\{a, c, d\} \setminus \{b, c\} = \{a, d\}$

symmetrische Differenz $A \Delta B = (A \setminus B) \cup (B \setminus A) = (A \cup B) \setminus (A \cap B)$
Beispiel: $\{a, c, d\} \Delta \{b, c\} = \{a, b, d\}$
Nachweis $(A \setminus B) \cup (B \setminus A) = (A \cup B) \setminus (A \cap B)$

bei gegebenem **Universum** U mit $A \subseteq U$:

Komplement $\overline{A} = U \setminus A = \{x \in U \mid \neg(x \in A)\}$
 $(x \in \overline{A}) \leftrightarrow \neg(x \in A)$
Beispiel: Für $U = \{a, b, c, d\}$ gilt $\overline{\{a, c\}} = \{b, d\}$

Wichtige Beziehungen zwischen Mengenoperationen

(analog Regeln für $\neg, \vee, \wedge$ in der Aussagenlogik)

Für alle Mengen A, B, C gilt

- ▶ $A \cup A = A, A \cap A = A$
- ▶ $A \cup \emptyset = A, A \cap \emptyset = \emptyset,$
- ▶ $A \cup B = B \cup A$ und $A \cap B = B \cap A$
(Kommutativität von $\cap$ und $\cup$)
- ▶ $A \cup (B \cup C) = (A \cup B) \cup C$
 $A \cap (B \cap C) = (A \cap B) \cap C$
(Assoziativität von $\cap$ und $\cup$)
- ▶ $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
 $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
(Distributivgesetze)
- ▶ $\overline{\overline{A}} = A$
- ▶ $\overline{A \cup B} = \overline{A} \cap \overline{B}$ und $\overline{A \cap B} = \overline{A} \cup \overline{B}$
(DeMorgansche Regeln)
- ▶ $\overline{\overline{A \cap B}} = A \cup B$ und $\overline{\overline{A \cup B}} = A \cap B$
(Dualität von $\cap$ und $\cup$)

(ÜA)

Disjunkte Mengen

Mengen A und B mit $A \cap B = \emptyset$ heißen **disjunkt**.

Beispiele:

- ▶ $\{a, b\}$ und $\{c, d\}$ sind disjunkt, weil $\{a, b\} \cap \{c, d\} = \emptyset$
- ▶ $\{a, b\}$ und $\{b, c\}$ sind nicht disjunkt, weil $\{a, b\} \cap \{b, c\} = \{b\} \neq \emptyset$
- ▶ Jede Menge M ist disjunkt zu $\emptyset$, weil $M \cap \emptyset = \emptyset$
- ▶ $2\mathbb{Z}$ und $2\mathbb{Z} + 1$ sind disjunkt
- ▶ $2\mathbb{Z}$ und $3\mathbb{Z}$ sind nicht disjunkt, weil $0 \in 2\mathbb{Z} \cap 3\mathbb{Z}$, also $2\mathbb{Z} \cap 3\mathbb{Z} \neq \emptyset$

Mengen $A_1, \dots, A_n$ heißen genau dann **paarweise disjunkt**, wenn $\forall i \in \{1, \dots, n\} \forall j \in \{1, \dots, n\} \setminus \{i\} : A_i \cap A_j = \emptyset$

Beispiele:

- ▶ $A_1 = \{a, d\}, A_2 = \{c\}, A_3 = \{b, e\}$ sind paarweise disjunkt, weil $A_1 \cap A_2 = \{a, d\} \cap \{c\} = \emptyset$, $A_1 \cap A_3 = \{a, d\} \cap \{b, e\} = \emptyset$ und $A_2 \cap A_3 = \{c\} \cap \{b, e\} = \emptyset$
- ▶ $A_1 = \{a, d\}, A_2 = \{c\}, A_3 = \{b, d\}$ sind nicht paarweise disjunkt, weil $A_1 \cap A_3 = \{a, d\} \cap \{b, d\} = \{d\} \neq \emptyset$

Zerlegungen (Partitionen) von Mengen

Eine Familie $\{A_i \mid i \in I\}$ paarweise disjunkter nichtleerer Mengen A_i mit

$$\bigcup_{i \in I} A_i = B$$

heißt **Zerlegung (Partition)** der Menge B .

Beispiele:

- ▶ $\underbrace{\{a, b\}}_{A_1}, \underbrace{\{c\}}_{A_2}$ ist eine Zerlegung der Menge $\{a, b, c\}$,

weil $A_1 \cap A_2 = \{a, b\} \cap \{c\} = \emptyset$ und
 $A_1 \cup A_2 = \{a, b\} \cup \{c\} = \{a, b, c\}$

- ▶ $\underbrace{\{a\}}_{A_1}, \underbrace{\{b\}}_{A_2}, \underbrace{\{c\}}_{A_3}$ ist eine Zerlegung der Menge $\{a, b, c\}$

- ▶ $\underbrace{\{a, b\}}_{A_1}, \underbrace{\emptyset}_{A_2}, \underbrace{\{c\}}_{A_3}$ ist keine Zerlegung der Menge $\{a, b, c\}$

- ▶ $\underbrace{\{a\}}_{A_1}, \underbrace{\{c\}}_{A_2}$ ist keine Zerlegung der Menge $\{a, b, c\}$,

aber eine Zerlegung der Menge $\{a, c\}$

- ▶ $\{2\mathbb{N}, 2\mathbb{N} + 1\}$ ist eine Zerlegung der Menge $\mathbb{N}$, $\{2\mathbb{N}, 3\mathbb{N}, 4\mathbb{N}\}$ nicht

Potenzmenge

Die **Potenzmenge** 2^A einer Menge A ist die Menge aller Teilmengen von A

$$2^A = \{B \mid B \subseteq A\}$$

Beispiele:

- ▶ Für $A = \{0\}$ gilt $2^A = 2^{\{0\}} = \{\emptyset, \{0\}\}$,
- ▶ Für $A = \{\heartsuit\}$ gilt $2^A = 2^{\{\heartsuit\}} = \{\emptyset, \{\heartsuit\}\}$,
- ▶ $2^{\{a,b,c\}} = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a,b\}, \{b,c\}, \{a,c\}, \{a,b,c\}\}$,
- ▶ Für $A = \{0\}$ gilt $2^{(2^A)} = 2^{(2^{\{0\}})} = \{\emptyset, \{\emptyset\}, \{\{0\}\}, \{\emptyset, \{0\}\}\}$,
- ▶ Amtssprachen in der Schweiz:
 $S = \{\text{Deutsch, Französisch, Italienisch, Rätoromanisch}\}$
mögliche Amtssprachkenntnisse der Bewohner: 2^S

Mächtigkeit der Potenzmengen endlicher Mengen:

$$\text{Für jede endliche Menge } A \text{ gilt } |2^A| = 2^{|A|}$$

Beweis durch Induktion über $|A|$

Modellierungsbeispiele mit Potenzmengen

- ▶ Menge aller Möglichkeiten, die Bauteile A, B, C, D einzusetzen (nicht notwendig alle, jedes höchstens einmal)
= Menge aller Teilmengen der Menge $\{A, B, C, D\}$
= $2^{\{A, B, C, D\}}$

Anzahl aller Möglichkeiten:

$$|2^{\{A, B, C, D\}}| = 2^{|\{A, B, C, D\}|} = 2^4 = 16$$

- ▶ Menge aller in einem Skatblatt aus 7 Karten möglichen Farbkombinationen
= Menge aller **nichtleeren** Teilmengen von $\{\diamond, \heartsuit, \spadesuit, \clubsuit\}$
= $2^{\{\diamond, \heartsuit, \spadesuit, \clubsuit\}} \setminus \{\emptyset\}$
($2^4 - 1 = 15$ Möglichkeiten)

Was bisher geschah

Modellierung von Aussagen
in klassischer Aussagen-Logik

Modellierung von Daten durch Mengen

- ▶ Darstellung:
 - extensional durch Angabe aller Elemente
(nur für endliche Mengen möglich)
 - intensional durch Angabe der charakterisierenden
Eigenschaft aller Elemente der Menge
(als prädikatenlogische Formel)
- ▶ leere Menge $\emptyset$
- ▶ Mengenbeziehungen $\in, \subseteq, =, \subset$
- ▶ Mengenoperationen $\cup, \cap, \overline{}, \setminus, \Delta$
- ▶ Mächtigkeit (Kardinalität) endlicher Mengen
- ▶ disjunkte Mengen, Zerlegungen
- ▶ Potenzmenge der Menge M : 2^M

Aussonderungsprinzip

Zu jeder Menge A und jeder Eigenschaft P existiert eine Menge

$$B = \{x \mid (x \in A) \wedge (x \text{ hat die Eigenschaft } P)\} = \{x \in A \mid P(x)\}$$

Beispiele:

- ▶ $A =$ Menge aller Fahrräder und ($P(x)$ gdw. x rosa)
 $B =$ Menge aller rosa Fahrräder
- ▶ $A =$ Menge aller Skatkarten und
 $P(x)$ gdw. x bedient ($\spadesuit$, A) (in Null-Spiel)
 $B = \{(\spadesuit, W) \mid W \in \{7, 8, 9, 10, B, D, K\}\}$
- ▶ $A = \{x \mid (x \in \mathbb{N}) \wedge (x \leq 100)\} = \{0, \dots, 100\}$ und
($P(x)$ gdw. x gerade)
 $B = \{x \mid (x \in \mathbb{N}) \wedge (x \leq 100) \wedge (2|x)\}$
- ▶ $A = 2\mathbb{N}$ und ($P(x)$ gdw. $x \leq 100$)
- ▶ $A = \mathbb{N}$ und ($P(x)$ gdw. $2|x$ und $x \leq 100$)

Russells Paradox

(Problem der naiven Mengenlehre)

Annahme:

Es existiert eine Menge A , die **alle** Mengen (als Elemente) enthält.

Nach Aussonderungsprinzip mit Eigenschaft $P(x)$ gdw. $\neg(x \in x)$ existiert dann auch die Menge

$$B = \{x \mid (x \in A) \wedge P(x)\} = \{x \mid (x \in A) \wedge \neg(x \in x)\}$$

Frage: **Gilt $B \in B$?**

Weil B eine Menge und jede Menge Element von A ist, gilt $B \in A$. ($\star$)

Für jede Menge x gilt $(x \in x)$ oder $\neg(x \in x)$, also zwei Fälle für $x = B$:

1. $(B \in B)$: nach Def. von B also $B \in A$ und $\neg(B \in B)$ (Widerspruch)
2. $\neg(B \in B)$: nach Def. von B also $\neg((B \in A) \wedge \neg(B \in B))$, wegen $\neg((B \in A) \wedge \neg(B \in B)) \equiv \neg(B \in A) \vee (B \in B)$, also zwei Fälle:
 - 2.1 $\neg(B \in A)$: Widerspruch zu $(B \in A)$ in ($\star$)
 - 2.2 $(B \in B)$: Widerspruch

Annahme führt in in allen Fällen zum Widerspruch, ist damit widerlegt.
Eine solche Menge A , die jede Menge (und damit insbesondere auch B) als Element enthält, kann nicht existieren.

alternative Formulierungen: Barbier, Kreter

Mögliche Lösungen

- ▶ axiomatischer Aufbau der Mengenlehre
Fundierungsaxiom:
Keine Menge kann sich selbst als Element enthalten.
- ▶ Klassentheorie (Mengen, Klassen, echte Klassen)
Klasse aller Mengen ist echte Klasse
- ▶ Typentheorie (Hierarchie)
alle Elemente einer Menge M haben niedrigeren Typ als M

(Kartesisches) Produkt von Mengen

$$A \times B = \{(x, y) \mid (x \in A) \wedge (y \in B)\}$$

Beispiele:

- ▶ $\{a, b\} \times \{1, 2, 3\} = \{(a, 1), (a, 2), (a, 3), (b, 1), (b, 2), (b, 3)\}$
- ▶ $\{1, \dots, m\} \times \{1, \dots, n\}$
 $= \{(i, j) \mid i \in \{1, \dots, m\} \wedge j \in \{1, \dots, n\}\}$
häufig als Indexmenge in Matrizen, digitalen Bildern,...
- ▶ Namen = Vornamen $\times$ Familiennamen
- ▶ Telefonbuch = (Namen $\times$ Vornamen) $\times$ Nummer
- ▶ Zeit = (Stunde $\times$ Minute) $\times$ Sekunde
 $= (\{0, \dots, 23\} \times \{0, \dots, 59\}) \times \{0, \dots, 59\}$

Für alle endlichen Mengen A und B gilt $|A \times B| = |A| \cdot |B|$

Beispiele

- ▶ $(1, 2) \in (\{0, 1\} \times \{1, 2\})$, aber $(1, 2) \notin (\{1, 2\} \times \{0, 1\})$
 $(\{0, 1\} \times \{1, 2\}) \cap (\{1, 2\} \times \{0, 1\}) = \{(1, 1)\}$
- ▶ Cafeteria-Essen aus
 $\{\text{Bratwurst, Wiener, Knacker}\} \times \{\text{Kartoffelsalat, Nudelsalat}\}$
 $3 \cdot 2 = 6$ Möglichkeiten
- ▶ $12 \cdot 15$ mögliche (gemischte) Tanzpaare aus
je einem von 12 Männern und je einer von 15 Frauen
- ▶ $\{a\} \times \mathbb{N} = \{(a, 0), (a, 1), (a, 2), \dots\} = \{a_0, a_1, a_2, \dots\}$

Eigenschaften des Produktes von Mengen

- ▶ $\times$ ist nicht kommutativ

z.B. $\{a, b\} \times \{1, 2\} = \{(a, 1), (a, 2), (b, 1), (b, 2)\}$
 $\neq \{1, 2\} \times \{a, b\} = \{(1, a), (2, a), (1, b), (2, b)\}$

- ▶ $\times$ ist „fast“ assoziativ,

$(A \times B) \times C$ und $A \times (B \times C)$ mit $A \times B \times C$ identifizierbar

z.B. $(\{a\} \times \{1, 2\}) \times \{\alpha, \beta\} = \{(a, 1), (a, 2)\} \times \{\alpha, \beta\}$
 $= \{((a, 1), \alpha), ((a, 2), \alpha), ((a, 1), \beta), ((a, 2), \beta)\}$
und $\{a\} \times (\{1, 2\} \times \{\alpha, \beta\}) = \{a\} \times \{(1, \alpha), (1, \beta), (2, \alpha), (2, \beta)\}$
 $= \{(a, (1, \alpha)), (a, (2, \alpha)), (a, (1, \beta)), (a, (2, \beta))\}$

lassen sich beide eineindeutig zuordnen zu

$$\{(a, 1, \alpha), (a, 2, \alpha), (a, 1, \beta), (a, 2, \beta)\} \subseteq A \times B \times C$$

- ▶ Distributivgesetze: Für alle Mengen A, B, C gilt

$$(A \cup B) \times C = (A \times C) \cup (B \times C) \text{ und}$$

$$(A \cap B) \times C = (A \times C) \cap (B \times C) \quad (\ddot{U}A)$$

- ▶ Für jede Menge A gilt $A \times \emptyset = \emptyset$.

- ▶ Für alle Mengen A, B, C gilt:

$$\text{Aus } A \subseteq B \text{ folgt } A \times C \subseteq B \times C$$

Modellierungsbeispiel Skatkarten

Formale Darstellung der Karten:

Jede Karte ist eindeutig repräsentiert durch

Farbe aus der Menge $F = \{\diamondsuit, \heartsuit, \spadesuit, \clubsuit\}$ und

Wert ► Zahl aus der Menge $Z = \{7, 8, 9, 10\}$ oder

► Bild aus der Menge $B = \{B, D, K, A\}$

Menge aller Werte $W = Z \cup B$

Menge aller Karten:

$$K = \{(x, y) \mid x \in F \wedge y \in W\} = F \times (Z \cup B)$$

Verteilung der Karten nach dem Geben:

► Skat: $S = \{k_{s,1}, k_{s,2}\} \subseteq K$ mit $|S| = 2$

► für jedes $i \in \{1, 2, 3\}$ Blatt des Spielers i :

$$B_i = \{k_{i,1}, \dots, k_{i,10}\} \subseteq K \text{ mit } |B_i| = 10$$

wobei

$$1. \quad B_1 \cap B_2 = \emptyset, B_1 \cap B_3 = \emptyset, B_2 \cap B_3 = \emptyset,$$

$$S \cap B_1 = \emptyset, S \cap B_2 = \emptyset, S \cap B_3 = \emptyset \text{ und}$$

$$2. \quad K = S \cup B_1 \cup B_2 \cup B_3$$

Also ist $\{S, B_1, B_2, B_3\}$ eine disjunkte Zerlegung von K .

Vereinigung disjunkter Mengen

Vereinigung $A \cup B = \{x \mid (x \in A) \vee (x \in B)\}$

Beispiel: Kunden $K = \{a, b, d\}$, Lieferanten $L = \{b, c, d\}$,

Vereinigung $G = K \cup L = \{a, b, c, d\}$ (Geschäftspartner)

enthält keine Information darüber, ob b Kunde oder Lieferant ist

Idee: Mengen vor Vereinigung durch Hinzufügen einer Kennzeichen-Komponente „disjunkt machen“

(Verwaltung von Paaren aus Kennzeichen und Element)

Disjunkte Vereinigung der Mengen $\{A_i\}_{i \in I}$ mit Indexmenge I :

$$\dot{\bigcup}_{i \in I} A_i = \bigcup_{i \in I} A'_i \quad \text{mit} \quad A'_i = \{i\} \times A_i = \{(i, x) \mid x \in A_i\}$$

Mächtigkeit der disjunkten Vereinigung der Mengen $\{A_i\}_{i \in I}$:

$$\left| \dot{\bigcup}_{i \in I} A_i \right| = \left| \bigcup_{i \in I} A'_i \right| = \sum_{i \in I} |A_i|$$

Beispiel Geschäftspartner

- ▶ Menge aller Kunden: $A_K = \{a, b, d\}$
- ▶ Menge aller Lieferanten: $A_L = \{b, c, d\}$,
- ▶ Kennzeichnung (Indexmenge) $I = \{K, L\}$
- ▶ gekennzeichnete Geschäftspartner:
Kunden: $A'_K = \{K\} \times A_K = \{(K, a), (K, b), (K, d)\}$,
Lieferanten: $A'_L = \{L\} \times A_L = \{(L, b), (L, c), (L, d)\}$
- ▶ Menge aller gekennzeichneten Geschäftspartner:
 $A'_K \cup A'_L = \{(K, a), (K, b), (L, b), (L, c), (K, d), (L, d)\}$

Anzahl der (gekennzeichneten) Geschäftspartner

$$|A_K \dot{\cup} A_L| = |A'_K \cup A'_L| = |A_K| + |A_L| = 6$$

Was bisher geschah

Modellierung von **Aussagen** in klassischer Aussagen-**Logik**

Modellierung von **Daten** durch **Mengen**

- ▶ extensionale und intensionale Darstellung
- ▶ Mächtigkeiten von (endlichen) Mengen $|M|$
- ▶ Beziehungen zwischen Mengen $\subseteq, =, \subset$
- ▶ leere Menge $\emptyset$
- ▶ Potenzmenge 2^M
- ▶ Mengen-Operationen $\cup, \cap, \overline{}, \setminus, \Delta$,
- ▶ Zerlegungen
- ▶ disjunkte Vereinigung
- ▶ (kartesisches) Produkt $\times$

Modellierungsbeispiele

Kartesisches Produkt mehrerer Mengen

$$\bigtimes_{i=1}^n A_i = A_1 \times \cdots \times A_n = \{(x_1, \dots, x_n) \mid \forall i \in \{1, \dots, n\} : x_i \in A_i\}$$

Für endlich viele endliche Mengen A_i gilt

$$\left| \bigtimes_{i=1}^n A_i \right| = \prod_{i=1}^n |A_i|$$

Beispiele:

- ▶ 5-Gänge-Menü: Auswahl aus
3 Vorspeisen, 2 Suppen, 5 Hauptgerichten, 3 Desserts, 2 Käse
 $3 \cdot 2 \cdot 5 \cdot 3 \cdot 2 = 180$ mögliche Kombinationen
- ▶ $|\{0, 1\}|^n = 2^n$ Binärwörter mit n Stellen
- ▶ 10^3 höchstens dreistellige Dezimalzahlen
- ▶ 6^5 verschiedene Ergebnisse beim fünfmal aufeinanderfolgenden Würfeln mit einem Würfel
- ▶ 6^5 verschiedene Ergebnisse beim Würfeln mit fünf verschiedenfarbigen Würfeln

Iterierte Produkte einer Menge

$$A^{\textcolor{red}{n}} = \bigtimes_{i=1}^n A$$

$$A^{\textcolor{red}{0}} = \{\varepsilon\} \quad \text{mit leerem Wort } \varepsilon$$

$$A^* = \bigcup_{n \in \mathbb{N}} A^n \quad (= A^0 \cup A^1 \cup \dots = \{\varepsilon\} \cup A^1 \cup \dots)$$

$$A^{\textcolor{red}{+}} = \bigcup_{n \in \mathbb{N} \setminus \{0\}} A^n = A^* \setminus \{\varepsilon\}$$

(Notation: statt ε mitunter auch $()$, $[]$)

Elemente aus A^n, A^+, A^* heißen (endliche)

- ▶ Folgen (Vektoren) $(a_1, a_2, \dots, a_n)$,
- ▶ Listen $[a_1, a_2, \dots, a_n]$, $[a_1, a_2, \dots]$ oder
- ▶ Wörter (Strings) $a_1 a_2 \dots a_n$

Iteriertes Produkt – Beispiele

- ▶ $\{0, 1\}^*$ Menge aller Binärwörter (beliebiger Länge)
z.B. 10, 10010, 010, ε
- ▶ Menge aller Binärwörter der Länge 3
 $\{0, 1\}^3 = \{000, 001, 010, 011, 100, 101, 110, 111\}$
- ▶ $\{0, 1, 2\}^*$ Menge aller Ternärwörter (beliebiger Länge)
z.B. 20, 1202010, ε
- ▶ Menge aller Ternärwörter der Länge 2
 $\{0, 1, 2\}^2 = \{00, 01, 02, 10, 11, 12, 20, 21, 22\}$
- ▶ $\{0\} \cup (\{1, 2, 3, 4, 5, 6, 7, 8, 9\} \times \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}^*)$
Menge aller natürlichen Zahlen in Dezimaldarstellung (ohne führende Nullen)
- ▶ $\{a, b\}^*$ alle Wörter (beliebiger Länge), die nur Buchstaben aus der Menge $\{a, b\}$ enthalten , z.B. *aba*, *ababa*, *aaaa*, *bbbaaa*, ε
- ▶ $\{\{a, b\}^*\}^*$ alle Folgen (Listen) solcher Wörter
z.B. $[a, aa, aaa]$, $[ba]$, ε , $[\varepsilon, \varepsilon, \varepsilon]$

Folgen

Folgen (Listen, Wörter) werden definiert:

extensional durch Angabe der **Elemente** und ihrer **Reihenfolge**
Beispiele: 3210, $[1, 4, 9, 16, 25]$, *abababababa*

intensional durch Angabe einer Eigenschaft, die für jeden Index i das i -te Element eindeutig bestimmt.
Beispiele: $(4 - i)_{1 \leq i \leq 4}$, $(i^2)_{i \in \{1, \dots, 5\}}$,

$$(w_i)_{0 \leq i \leq 10} \text{ mit } w_i = \begin{cases} a & \text{falls } i \in 2\mathbb{Z} \\ b & \text{sonst} \end{cases}$$

$$(v_i)_{i \in \mathbb{N}} \text{ mit } v_i = \begin{cases} a & \text{falls } i \in 2\mathbb{Z} \\ b & \text{sonst} \end{cases}$$

$$(i^2)_{i \in \mathbb{N}} = [0, 1, 4, 9, \dots], (3)_{k \in \mathbb{N}} = [3, 3, 3, 3, \dots]$$

Länge der Folge $(a_n)_{n \in I}$:

Anzahl der Elemente (= Mächtigkeit der Indexmenge $I \subseteq \mathbb{N}$)

Modellierung durch Mengen und Folgen

- ▶ **Menge** aller Skatkarten in einem Spiel:
 $S = \{\diamondsuit, \heartsuit, \spadesuit, \clubsuit\} \times (\{7, 8, 9, 10\} \cup \{B, D, K, A\})$
- ▶ **Folge** aller Unter (B) nach Wert aufsteigend geordnet
 $[(\clubsuit, B), (\spadesuit, B), (\heartsuit, B), (\diamondsuit, B)] \in S^4$
- ▶ **Folge** aller Karten der Farbe $\spadesuit$ nach Wert (Nullspiel) aufsteigend geordnet
 $[(\spadesuit, 7), (\spadesuit, 8), (\spadesuit, 9), (\spadesuit, 10), (\spadesuit, B), (\spadesuit, D), (\spadesuit, K), (\spadesuit, A)] \in S^8$
- ▶ **Menge** aller Möglichkeiten der (zwei) Karten im Skat:
 $\{\{h, k\} \mid h \in S \wedge k \in S \wedge h \neq k\} = \{M \subseteq S \mid 2 = |M|\} \subseteq 2^S$
(Warum nicht $\subseteq S^2$?)
- ▶ Blatt = **Menge** aller Karten auf der Hand (zu Beginn des Spieles)
 $B \in 2^S$ mit $|B| = 10$ (10 verschiedene Karten)
- ▶ Kartenfächer (zu Beginn des Spieles): **Folge** $(k_1, \dots, k_{10}) \in S^{10}$ mit $|\{k_1, \dots, k_{10}\}| = 10$ (alle Karten verschieden), z.B.
 $[(\spadesuit, B), (\heartsuit, B), (\diamondsuit, A), (\diamondsuit, 9), (\clubsuit, A), (\clubsuit, K), (\clubsuit, 9), (\heartsuit, 10), (\heartsuit, 8), (\heartsuit, 7)]$
- ▶ Stich: **Folge** von drei nacheinander gelegten Karten (geordnetes Tripel) $(k_1, k_2, k_3) \in S^3$ mit $|\{k_1, \dots, k_3\}| = 3$
- ▶ Spiel: **Folge** der (während des Spiels gefallenen) Stiche $\in (S^3)^{10}$

Zusammenhänge Folgen – Mengen

zu gegebener Folge $(a_n)_{n \in I}$:

- ▶ Menge $\{a_n \mid n \in I\}$ der Elemente der Folge $(a_n)_{n \in I}$ (eindeutig)
 - ▶ Folge $[1, 4, 9, 16]$, Menge der Elemente $\{1, 4, 9, 16\}$
 - ▶ Folge $[a, a, a, \dots]$, Menge der Elemente $\{a\}$

(Gegebene Folge nicht eindeutig rekonstruierbar)

- ▶ Menge aller Anfangsstücke der Folge (eindeutig)
Beispiel: $(2^n)_{n \in \mathbb{N}}$, Menge $\{\varepsilon, [1], [1, 2], [1, 2, 4], \dots\}$
(Folge lässt sich eindeutig rekonstruieren.)

zu gegebener Menge A :

- ▶ Folgen $(a_n)_{n \in I}$ sind durch Mengen definiert (A^*, A^ω)
- ▶ beliebige Anordnung der Elemente einer Menge zu Folgen (i.A. nicht eindeutig, mehrere Möglichkeiten), z.B.
 - ▶ $M = \{a, b, c, d\}$, Folgen $[a, b, c, d], [b, d, b, c, d, a]$
 - ▶ $M = \mathbb{N}$, Folgen $[0, 1, 2, \dots]$, $[0, 2, 4, 6, 8, 1, 3, 5, 7, 9, 10, 12, \dots]$
 - ▶ $M = \mathbb{Z}$, Folgen $[0, 1, -1, 2, -2, 3, -3, \dots]$
 $[0, 1, 2, 3, -3, -2, -1, 4, 5, \dots]$

Alphabet, Wort, (formale) Sprache

Alphabet (endliche) Menge A von Symbolen

Wort endliche Folge von Symbolen $w = w_1 \cdots w_n$ mit
 $\forall i \in \{1, \dots, n\} : w_i \in A$

Länge eines Wortes $|w|$ = Anzahl der Symbole in w

Anzahl der Vorkommen eines Symbolen in einem Wort

$|w|_a$ = Anzahl der a in w (für $a \in A$)

Sprache Menge von Wörtern $L \subseteq A^*$

Wörter – Beispiele

abbaavatar

ist ein Wort (Zeichenkette) mit Symbolen aus der Menge $\{a, b, r, t, v\}$,
tar und *bratbart* auch, *trallala* und *ab + ba* nicht

2024

ist ein Wort mit Symbolen aus der Menge $\{0, 4, 2\}$,
024 und 0420040 auch, -40 nicht

$(x + y) \cdot (z - x)$

ist ein Wort mit Symbolen aus der Menge $\{x, y, z, (,), +, -, \cdot\}$,
 $()xz(xy + -$ auch, $x + 3 \cdot z$ nicht

$(\neg p \wedge p) \rightarrow q$

ist ein Wort mit Symbolen aus der Menge $\{p, q, \wedge, \neg, \rightarrow, (,)\}$,
 $q \rightarrow (p \rightarrow q)$ und $\wedge)(\neg p \wedge$ auch, $p \leftrightarrow q$ nicht

otto holt obst .

ist ein Wort mit Symbolen aus der Menge $\{otto, obst, holt, .\}$,
. otto . . otto auch, los otto nicht

Verkettung von Wörtern (Folgen)

Verkettung $\circ$ von **Wörtern**: $\circ : A^* \times A^* \rightarrow A^*$

Für alle Wörter $u, v \in A^*$ ist $u \circ v = w$ mit

$$\forall i \in \{1, \dots, |u| + |v|\} : w_i = \begin{cases} u_i & \text{falls } i \leq |u| \\ v_{i-|u|} & \text{falls } i > |u| \end{cases}$$

(Für $u = u_1 \cdots u_{|u|}, v = v_1 \cdots v_{|v|} \in A^*$ ist $u \circ v = u_1 \cdots u_{|u|} v_1 \cdots v_{|v|}$)

Beispiel: $\text{anne} \circ \text{marie} = \text{annemarie}$

Eigenschaften der Operation $\circ$:

- ▶ $\circ$ ist assoziativ, d.h.

$$\forall u \in A^* \forall v \in A^* \forall w \in A^* ((u \circ v) \circ w = u \circ (v \circ w))$$

- ▶ $\forall w \in A^* (\varepsilon \circ w = w \circ \varepsilon = w)$
(Das leere Wort ε ist neutrales Element für $\circ$)
- ▶ $\circ$ ist nicht kommutativ.

Gegenbeispiel: $u = \text{marie}, v = \text{anne}$

$u \circ v = \text{marianne} \neq \text{annemarie} = v \circ u$

Umkehrung (gespiegeltes Wort)

Umkehrung w^R von w definiert durch

$$\forall i \in \{1, \dots, |w|\} : w_i^R = w_{|w|-i+1}$$

(Für $w = w_1 \cdots w_{|w|}$ ist $w^R = w_{|w|} \cdots w_1$)

Beispiele: $(marie)^R = eiram$, $(2023)^R = 3202$, $(101)^R = 101$

$$\left(to \circ \left(\left(m \circ (ate)^R \right)^R \circ n \right)^R \right)^R = \dots$$

Fakt

Für jedes Wort $w \in A^*$ gilt $(w^R)^R = w$.

Palindrom

Palindrom: Wort w mit $w = w^R$

B: *anna*, *neben*, ε , jedes Wort der Länge 1

Die Menge aller Palindrome über dem Alphabet A ist

$$\begin{aligned} L_{\text{pal}} &= \{w \in A^* \mid w = w^R\} \\ &= \underbrace{\{w \circ w^R \mid w \in A^*\}}_{L_{\text{pal}0}} \cup \underbrace{\{w \circ a \circ w^R \mid w \in A^* \wedge a \in A\}}_{L_{\text{pal}1}} \end{aligned}$$

Beispiele für Wörter aus L_{pal} :

- ▶ $otto = ot \circ to = ot \circ (ot)^R$ für $w = ot \in A^* = \{a, \dots, z\}^*$
- ▶ $reliefpfeiler = relief \circ p \circ feiler = relief \circ p \circ (relief)^R$
für $w = relief \in A^* = \{a, \dots, z\}^*$
- ▶ $1 = \varepsilon \circ 1 \circ \varepsilon = \varepsilon \circ 1 \circ \varepsilon^R$ für $A = \{0, 1\}$
- ▶ $\varepsilon = \varepsilon \circ \varepsilon = \varepsilon \circ \varepsilon^R$

Was bisher geschah

Modellierung von Aussagen durch Logiken

Modellierung von Daten durch

- Mengen
- ▶ extensionale und intensionale Darstellung
 - ▶ Mächtigkeiten endlicher Mengen, $\emptyset$
 - ▶ Beziehungen zwischen Mengen $\subseteq, =, \subset$
 - ▶ Mengen-Operationen $\cup, \cap, \overline{}, \setminus, \Delta, \times, {}^n, *, 2^M$

- Folgen (Vektoren, Listen, Wörtern) über einer Menge A
- ▶ extensionale und intensionale Darstellung
 - ▶ Länge von Folgen, leeres Wort ε (leere Folge)
 - ▶ unendliche Folgen,
endliche Folgen fester Länge (Tupel, Vektoren),
endliche Folgen variabler Länge (Wörter, Listen)
 - ▶ Operationen auf Wörtern: Verkettung $\circ, {}^R$
 - ▶ Palindrome

Präfix-Beziehung zwischen Wörtern (Folgen)

Präfix (Anfangswort) $\sqsubseteq$

$$\forall u \in A^* \forall v \in A^* ((u \sqsubseteq v) \leftrightarrow (\exists w \in A^* (u \circ w = v)))$$

(Für zwei beliebige Wörter $u \in A^*$, $v \in A^*$ gilt $u \sqsubseteq v$ genau dann, wenn ein Wort $w \in A^*$ existiert, so dass $u \circ w = v$ gilt.)

Beispiele:

- ▶ $an \sqsubseteq anna$ (mit $w = na$)
- ▶ $tom \sqsubseteq tomate$ (mit $w = ate$)
- ▶ $oma \not\sqsubseteq tomate$
- ▶ $tat \not\sqsubseteq tomate$
- ▶ für jedes Wort $u \in A^*$ gilt $\varepsilon \sqsubseteq u$ (mit $w = u$)
- ▶ für jedes Wort $u \in A^*$ gilt $u \sqsubseteq u$ (mit $w = \varepsilon$)

(analog zur Teiler-Beziehung zwischen natürlichen Zahlen)

Postfix-Beziehung auf Wörtern (Folgen)

Postfix (Suffix):

$$\forall u \in A^* \forall v \in A^* \quad (\text{Postfix}(u, v) \leftrightarrow (\exists w \in A^* (w \circ u = v)))$$

Für zwei Wörter $u = u_1 \cdots u_m \in A^*$, $v = v_1 \cdots v_n \in A^*$ heißt u genau dann Postfix (Suffix) von v , wenn ein Wort $w \in A^*$ existiert, so dass $w \circ u = v$ gilt.

Beispiele:

- ▶ *enten* ist Postfix von *studenten* (mit $w = \textit{stud}$)
- ▶ ε ist Postfix von *studenten* (mit $w = \textit{studenten}$)
- ▶ *ente* ist kein Postfix von *studenten*
- ▶ *den* ist kein Postfix von *studenten*

Infix-Beziehung auf Wörtern (Folgen)

Infix (Teilwort, Faktor):

$$\forall u \in A^* \forall v \in A^* \left(\text{Infix}(u, v) \leftrightarrow (\exists w \in A^* \exists w' \in A^* (w \circ u \circ w' = v)) \right)$$

Für zwei Wörter $u = u_1 \cdots u_m \in A^*$, $v = v_1 \cdots v_n \in A^*$ heißt u genau dann Infix von v , wenn zwei Wörter $w, w' \in A^*$ existieren, so dass $w \circ u \circ w' = v$ gilt.

Beispiele:

- ▶ *oma* ist Infix von *tomate* (mit $w = t$, $w' = te$)
- ▶ *tom* ist Infix von *tomate* (mit $w = \varepsilon$, $w' = ate$)
- ▶ *den* ist Infix von *studenten* (mit $w = stu$, $w' = ten$)
- ▶ *ente* ist Infix von *studenten* (mit $w = stud$, $w' = n$)
- ▶ *ente* ist kein Infix von *student*
- ▶ *enten* ist Infix von *studenten* (mit $w = stud$, $w' = \varepsilon$)

Weitere Beziehungen zwischen Wörtern

bei gegebener Reihenfolge $<$ auf dem Alphabet A :

lexikographische Ordnung auf A^* :

$\forall u, v \in A^* : u \leq_{\text{lex}} v$ gdw.

L1: $u \sqsubseteq v$ oder

L2: $\exists w \in A^* \exists x, y \in A : x < y \wedge wx \sqsubseteq u \wedge wy \sqsubseteq v$

quasi-lexikographische (Längen-lexikographische) Ordnung auf A^* :

$\forall u, v \in A^* : u \leq_{\text{qllex}} v$ gdw.

Q1: $|u| < |v|$ oder

Q2: $|u| = |v| \wedge u \leq_{\text{lex}} v$

Beispiele: für $A = \{b, o, y\}$ mit $b < o < y$

► $bob \leq_{\text{lex}} bobby$ wegen $bob \sqsubseteq bobby$ (L1)

► $bob \leq_{\text{qllex}} bobby$ wegen $|bob| < |bobby|$ (Q1)

► $bobby \leq_{\text{lex}} bobo$ mit $w = bob, x = b, y = o$ (L2)

► $bobo \leq_{\text{qllex}} bobby$ wegen $|bobo| < |bobby|$ (Q1)

► $bob \leq_{\text{qllex}} boy$ wegen $|bob| = |boy|$ und $bob \leq_{\text{lex}} boy$ (Q2)

(Formale) Sprachen

Alphabet endliche Menge A von Symbolen

Wort über A : $w \in A^*$ (Folge von Symbolen aus A)

Sprache über A : $L \subseteq A^*$ (Menge von Wörtern über A)

Voraussetzung für maschinelle Verarbeitung:

endliche Darstellung von (evtl. unendlichen) Sprachen

verschiedene Darstellungen in den LV zur theoretischen Informatik
z.B. Automaten und formale Sprachen im 4. Semester (INB)

Beispiele für Sprachen

- ▶ Menge aller englischen Wörter $L_1 \subset \{a, \dots, z\}^*$
- ▶ Menge aller deutschen Wörter $L_2 \subset \{a, \dots, z, \text{ß}, \text{ä}, \text{ö}, \text{ü}\}^*$
- ▶ Menge aller möglichen RNA $L_3 \subseteq \{A, U, G, C\}^*$
- ▶ Menge aller natürlichen Zahlen in Dezimaldarstellung $L_4 = \{0, \dots, 9\}^*$ (evtl. mit führenden Nullen)
- ▶ Menge aller natürlichen Zahlen in Binärdarstellung (Bitfolgen beliebiger Länge) $L_5 = \{0, 1\}^*$
- ▶ Menge aller aussagenlogischen Formeln in $AL(\{p, q, r\})$
 $L_6 \subset \{p, q, r, \text{⊥}, \text{⊢}, \neg, \vee, \wedge, \rightarrow, \leftrightarrow, (,)\}^*$,
- ▶ Menge aller arithmetischen Ausdrücke über $\mathbb{Z}$ (ohne Variablen)
 $L_7 \subset \{0, \dots, 9, +, \cdot, -, /, (,)\}^*$,
- ▶ Menge aller deutschen Sätze $L_8 \subset (L_2 \cup \{., , !, ?, (,), -\})^*$

Sprachen als Mengen

Sprachen $L \subseteq A^*$ sind **Mengen** von Wörtern (endlichen Folgen)
Mengenbeziehungen auf Sprachen:

$$L \subseteq L' \quad \text{gdw.} \quad \forall w \in A^* ((w \in L) \rightarrow (w \in L'))$$

$$L = L' \quad \text{gdw.} \quad \forall w \in A^* ((w \in L) \leftrightarrow (w \in L'))$$

Mengenoperationen auf Sprachen:

$$L \cup L' = \{w \mid w \in L \vee w \in L'\}$$

$$L \cap L' = \{w \mid w \in L \wedge w \in L'\}$$

$$L \setminus L' = \{w \mid w \in L \wedge w \notin L'\}$$

Komplement einer Sprache $L \subseteq A^*$: $\bar{L} = A^* \setminus L$

Beispiel:

$$L = \{w \mid w \in \{0,1\}^* \wedge \exists n \in \mathbb{N}(|w| = 2n)\}$$

$$\bar{L} = \{w \mid w \in \{0,1\}^* \wedge \exists n \in \mathbb{N}(|w| = 2n + 1)\}$$

Operationen auf Sprachen

Verkettung $\circ$ von Sprachen:

$$L_1 \circ L_2 = \{u \circ v \mid (u \in L_1) \wedge (v \in L_2)\}$$

Beispiel:

$$\begin{aligned} L_1 &= \{111, 1, 10\} & L_2 &= \{00, 0\} \\ L_1 \circ L_2 &= \{111, 1, 10\} \circ \{00, 0\} \\ &= \{1110, 11100, 10, 100, 1000\} \end{aligned}$$

Spiegelung $L^R = \{w^R \mid w \in L\}$

Beispiel: $L = \{a, ab, aba, abab\}$
 $L^R = \{a, ba, aba, baba\}$

Iterierte Verkettung

- für **Sprachen** $L \subseteq A^*$

$$L^0 = \{\varepsilon\} \quad \forall n \in \mathbb{N}: \quad L^{n+1} = L \circ L^n = \underbrace{L \circ \dots \circ L}_{n+1\text{-mal}}$$

$$L^* = \bigcup_{n \in \mathbb{N}} L^n$$

$$L^+ = \bigcup_{n \in \mathbb{N} \setminus \{0\}} L^n$$

- für **Wörter** (endliche Folgen) $u \in A^*$:

$$u^n = \underbrace{u \circ \dots \circ u}_{n\text{-mal}} \in A^*,$$

$$u^* = \{u\}^* = \{u^n \mid n \in \mathbb{N}\} \subseteq A^*$$

$$u^+ = \{u\}^+ = \{u^n \mid n \in \mathbb{N} \setminus \{0\}\} \subseteq A^*$$

Beispiele:

$$(101)^3 = 101101101 \quad \text{und} \quad 101^3 = 10111$$

$$a^* = \{a^i \mid i \in \mathbb{N}\} = \{\varepsilon, a, aa, aaa, \dots\}$$

$$(ab)^* = \{(ab)^i \mid i \in \mathbb{N}\} = \{\varepsilon, ab, abab, ababab, \dots\}$$

$$\varepsilon^* = \{\varepsilon\} = \varepsilon^+$$

Mehr Beispiele für Sprachen

► $\{aa, b\}^*$

$$= \{u_1 \circ \dots \circ u_n \mid n \in \mathbb{N} \wedge \forall i \in \{1, \dots, n\} (u_i \in \{aa, b\})\}$$

$$= \{\varepsilon, b, aa, bb, aab, baa, bbb, aaaa, aabb, baab, bbaa, bbbb, \dots\}$$

$$ab \notin \{aa, b\}^*, ba \notin \{aa, b\}^*, aba \notin \{aa, b\}^*$$

► $\emptyset \circ \{aba, bb\} = \emptyset$

► $\{\varepsilon\} \circ \{aba, bb\} = \{aba, bb\}$

► $\{bb\}^* = \{w \in \{b\}^* \mid |w| \in 2\mathbb{N}\} = \{\varepsilon, bb, bbbb, \dots\}$

► $(\{1\} \circ \{0\}^*)^* = \{w \in \{0, 1\}^* \mid w_1 = 1\} \cup \{\varepsilon\}$

Reguläre Ausdrücke – Syntax

Die Menge $\text{RegExp}(A)$ aller **regulären Ausdrücke** über einem Alphabet A ist (induktiv) definiert durch:

- IA: $\emptyset \in \text{RegExp}(A)$,
 $\varepsilon \in \text{RegExp}(A)$ und
für jedes Symbol $a \in A$ gilt $a \in \text{RegExp}(A)$
- IS: für alle $E \in \text{RegExp}(A)$ und $F \in \text{RegExp}(A)$ gilt
 $(E + F), EF, (E)^* \in \text{RegExp}(A)$.

(Baumdarstellung)

Beispiele:

- ▶ $(\emptyset + 1) \in \text{RegExp}(\{0, 1\})$,
- ▶ $(\varepsilon + ((ab)^*a)^*) \in \text{RegExp}(\{a, b\})$,
- ▶ $(\spadesuit \heartsuit^* \emptyset + \diamond)^* \in \text{RegExp}(\{\diamond, \heartsuit, \spadesuit, \clubsuit\})$,
- ▶ für beliebiges Alphabet A gilt $\varepsilon \in \text{RegExp}(A)$,
 $(\varepsilon + \emptyset) \in \text{RegExp}(A)$,
- ▶ $0 + (1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9)(0 + 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9)^* \in \text{RegExp}(\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\})$

Reguläre Ausdrücke – Semantik

Jeder reguläre Ausdruck $E \in \text{RegExp}(A)$ repräsentiert die wie folgt definierte Sprache $L(E) \subseteq A^*$:

- IA:
- ▶ $L(\emptyset) = \emptyset$
 - ▶ $L(\varepsilon) = \{\varepsilon\}$
 - ▶ $\forall a \in A : L(a) = \{a\}$
- IS: $\forall E, F \in \text{RegExp}(A) :$
- ▶ $L(E + F) = L(E) \cup L(F)$
 - ▶ $L(EF) = L(E) \circ L(F)$
 - ▶ $L(E^*) = L(E)^*$

Eine Sprache $L \subseteq A^*$ heißt genau dann **regulär**, wenn ein regulärer Ausdruck $E \in \text{RegExp}(A)$ existiert, so dass $L = L(E)$.

Beispiele: Für $A = \{a, b\}$ gilt

$$\begin{aligned}L(ab^*) &= \{a, ab, abb, abbb, abbbb, \dots\} = \{ab^i \mid i \in \mathbb{N}\} \\L((ab)^*) &= \{\varepsilon, ab, abab, ababab, \dots\} = \{(ab)^i \mid i \in \mathbb{N}\} \\L((a + b)^*) &= \{a, b\}^* \\L(a^*b^*) &= \{u \circ v \mid u \in a^* \wedge v \in b^*\} = \{a^i b^j \mid i \in \mathbb{N} \wedge j \in \mathbb{N}\} \\L((a^*b^*)^*) &= \{a, b\}^*\end{aligned}$$

Beispiele

- ▶ Für $E = 0 + (1 + 2 + \dots + 9)(0 + 1 + \dots + 9)^*$ ist $L(E)$ = Menge aller Dezimaldarstellungen natürlicher Zahlen
- ▶ für $A = \{A, B, \dots, Z, a, b, \dots, z\}$ ist $L(A^*(oma + otto)A^*)$ = Menge aller Wörter mit Infix *oma* oder *otto* (oder beiden)
z.B. $tomate \in L(A^*(oma + otto)A^*)$, $lotto \in L(A^*(oma + otto)A^*)$,
 $ottomane \in L(A^*(oma + otto)A^*)$
- ▶ Menge aller möglichen HTWK-Email-Adressen
 $\subset L((A + \dots + Z + a + \dots + z + 0 + \dots + 9 + . + -)^* @htwk-leipzig.de)$

Reguläre Ausdrücke ermöglichen eine
endliche Darstellung **unendlicher** Sprachen.

Aber: Nicht jede (unendliche) Sprache ist regulär. (Warum?)

(mehr dazu in den LV zur theoretischen Informatik,
z.B. Automaten und Formale Sprachen im 4. Semester)

Beispiele regulärer Sprachen

- ▶ $L_1 = \emptyset$ ist eine reguläre Sprache, weil
 $\emptyset \in \text{RegExp}(A)$ und $L(\emptyset) = \emptyset = L_1$
- ▶ $L_2 = \{a, bab\}$ ist eine reguläre Sprache, weil
 $a + bab \in \text{RegExp}(A)$ und $L(a + bab) = L_2$
- ▶ $L_3 = \{w \in \{a, b, c\}^* \mid abba \sqsubseteq w\}$ ist regulär, weil
 $abba(a + b + c)^* \in \text{RegExp}(\{a, b, c\})$ und
 $L(abba(a + b + c)^*) = L_3$
- ▶ $L_4 = \{w \in \{a, b, c\}^* \mid aa \text{ oder } bbb \text{ sind Infix von } w\}$
ist eine reguläre Sprache, weil
 $(a + b + c)^*(aa + bbb)(a + b + c)^* \in \text{RegExp}(\{a, b, c\})$
und $L((a + b + c)^*(aa + bbb)(a + b + c)^*) = L_4$
- ▶ $\{w \in \{a, b\}^* \mid |w| \in 2\mathbb{N}\} = L(((a + b)(a + b))^*)$

Was bisher geschah

Modellierung von

- ▶ **Aussagen** durch Logiken:
 - ▶ klassische Aussagenlogik
 - ▶ klassische Prädikatenlogik (bisher nur Syntax):
(Mengen von) Individuen, Eigenschaften, Beziehungen
- ▶ **Daten** durch
 - ▶ Mengen
 - ▶ Folgen (Listen, Wörter)
 - ▶ Sprachen
(endlichen Beschreibung durch reguläre Ausdrücke)

(Modellierung der Individuen und Mengen von Individuen)

Relationen

Relationen repräsentieren **Zusammenhänge** zwischen Individuen

Definition

Jede Menge $R \subseteq A \times B$ heißt **Relation** zwischen den Mengen A und B .

Beispiele:

- ▶ $H \subseteq \text{Studenten} \times \text{Dozenten}$ mit $(s, d) \in H$ gdw. s hört (wenigstens) eine Vorlesung bei d .
- ▶ $S \subseteq \text{Skatkarten} \times \text{Skatkarten}$ mit $(h, k) \in S$ gdw. Karte h hat höheren Wert als Karte k
- ▶ $B \subseteq \text{Spieler} \times \text{Skatkarten}$ mit $(s, k) \in B$ gdw. Spieler s hat die Karte k auf der Hand.
- ▶ $R \subseteq \{a, b, c\} \times \{1, 2, 3, 4\}$ mit $R = \{(a, 2), (a, 3), (c, 2), (c, 4)\}$
- ▶ $R \subseteq \mathbb{R}^2$ mit $R = \{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 = 1\}$
- ▶ $\leq \subseteq \mathbb{R}^2$ mit $\leq = \{(m, n) \in \mathbb{R}^2 \mid m \leq n\}$
- ▶ $\emptyset \subseteq A \times B$ (leere Relation) für beliebige Mengen A und B .

Zur Definition einer Relation $R \subseteq A \times B$ gehören (sind anzugeben):

- ▶ die Mengen A, B (**Typ** der Relation) und
- ▶ die Menge R (extensional oder intensional)

Darstellung von Relationen

Darstellungsformen für Relation $R \subseteq A \times B$ auf **endlichen** Mengen A und B :

- ▶ Menge geordneter Paare (intensional oder extensional)
- ▶ für $A = \{a_1, \dots, a_m\}$ und $B = \{b_1, \dots, b_n\}$
 $(|A| \times |B|)$ -**Matrix** M mit Einträgen $m_{ij} \in \{0, 1\}$, wobei

$$\forall i \in \{1 \dots |A|\} \forall j \in \{1 \dots |B|\} : m_{ij} = \begin{cases} 1 & \text{falls } (a_i, b_j) \in R \\ 0 & \text{sonst} \end{cases}$$

- ▶ Diagramm (gerichteter Graph $G = (A \cup B, R)$)
mit Eckenmenge $A \cup B$ und Kantenmenge R

falls (wenigstens) eine der Mengen A und B **unendlich** ist,
i.A. intensionale Darstellung von Relationen $R \subseteq A \times B$ nötig

Relationen verschiedener Stelligkeit

Relation $R \subseteq A \times B$ für spezielle Mengen B

$B = A$, also $A \times B = A^2$

$R \subseteq A^2$ heißt **binäre (zweistellige) Relation auf A** .

Beispiele: $\leq \subseteq \mathbb{R}^2$ mit $\leq = \{(x, y) \in \mathbb{R}^2 \mid x \leq y\}$,

Identität auf A : $I_A \subseteq A^2$ mit $I_A = \{(x, x) \mid x \in A\}$

$B = A^{n-1}$, also $A \times B = A \times A^{n-1} = A^n$

$R \subseteq A^n$ heißt **n -stellige Relation auf A** .

Beispiel: $R = \{(x, y, z) \in \mathbb{N}^3 \mid x^2 + y^2 = z^2\} \subseteq \mathbb{N}^3$

$B = A^0 = \{\varepsilon\}$, also $A \times B = A \times \{\varepsilon\} = \{(a, \varepsilon) \mid a \in A\}$

Relation entspricht einer Teilmenge $R \subseteq A$.

$R \subseteq A$ heißt dann auch **einstellige Relation auf A** .

(Eigenschaft von Elementen aus A)

Beispiel: $G \subseteq \mathbb{N}$ mit $G = \{n \in \mathbb{N} \mid n > 5\}$

Anzahl von Relationen auf endlichen Mengen

Wiederholung: Für endliche Mengen A und B gilt

$$|A \times B| = |A| \cdot |B|$$

Menge aller Relationen zwischen A und B :

$$\{R \mid R \subseteq A \times B\} = 2^{A \times B}$$

Wieviele verschiedene Relationen (Teilmengen) $R \subseteq A \times B$?

$$2^{|A \times B|} = 2^{|A| \cdot |B|}$$

Spezialfälle:

$|B| = 1$: $2^{|A| \cdot |B|} = 2^{|A|}$ einstellige Relationen auf A
(Teilmengen von A).

$A = B$: $2^{|A \times A|} = 2^{|A|^2}$ binäre Relationen auf A .

$B = A^{n-1}$: $2^{|A| \cdot |A|^{n-1}} = 2^{|A|^n}$ n -stellige Relationen auf A .

Häufige Anwendungen

- ▶ Ontologien, z.B. Komponenten eines Systems:
istTeilVon (Hand, Arm), istTeilVon (Finger, Hand),
istEin (Daumen, Finger)
- ▶ Abläufe mit Übergangsrelation T zwischen Zuständen ,
z.B. Münzspiel (aus der ersten Vorlesung): $T \subseteq \left(\{0, \dots, 4\}^4 \right)^2$
 $(0230, 1040) \in T$, $(0230, 0311) \in T$, $(0311, 1121) \in T$,
 $(1040, 1121) \in T$, $(1121, 1202) \in T$
- ▶ Datenbanken: jede Tabelle repräsentiert eine Relation T , z.B.

Id	Vorname	Name	Studieng.	Jg.	Gruppe
34567	Lisa	Klein	INB	25	3
12345	Erwin	Meier	MIB	25	1
56789	Paula	Richter	INB	23	2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

mit $T \subseteq \{0, 1, \dots, 9\}^5 \times \text{Vornamen} \times \text{Namen}$
 $\times \{\text{INB}, \text{MIB}\} \times \{0, 1, \dots, 9\}^2 \times \{1, 2, 3\}$

und z.B. $(12345, \text{Erwin}, \text{Meier}, \text{MIB}, 25, 1) \in T$

Abgeleitete Relationen

Inverse der binären Relation $R \subseteq A \times B$:

$$R^{-1} \subseteq B \times A \quad \text{mit} \quad R^{-1} = \{(b, a) \mid (a, b) \in R\}$$

Beispiele: $A = \{1, 2, 3, 4\}$, $B = \{a, b, c, d\}$

$$\{(1, a), (1, b), (2, b), (4, a)\}^{-1} = \{(a, 1), (b, 1), (b, 2), (a, 4)\},$$

$$\text{istKindVon}^{-1} = \text{istElternteilVon}, \text{istTeilVon}^{-1} = \text{hatTeil}, \leq^{-1} = \geq$$

Einschränkung der Relation $R \subseteq A \times B$ auf $M \subseteq A$:

$$R|_M \subseteq A \times B \quad \text{mit} \quad R|_M = \{(a, b) \in R \mid a \in M\}$$

Beispiele: $\{(1, a), (1, b), (2, b), (4, a)\}|_{\{2,3,4\}} = \{(2, b), (4, a)\}$,

$$\text{istElternteilVon}|_{\text{Frauen}} = \text{istMutterVon},$$

Projektionen der Relation $R \subseteq A \times B$:

$$\pi_1(R) \subseteq A \quad \text{mit} \quad \pi_1(R) = \{a \in A \mid (a, b) \in R\}$$

$$\pi_2(R) \subseteq B \quad \text{mit} \quad \pi_2(R) = \{b \in B \mid (a, b) \in R\}$$

Beispiele: $\pi_1(\{(1, a), (1, b), (2, b), (4, a)\}) = \{1, 2, 4\}$,

$$\pi_2(\text{istKindVon}) = \text{Menge aller Eltern}$$

Mengenoperationen auf Relationen

Relationen sind Mengen.

Für Relationen $R \subseteq A \times B$, $S \subseteq A' \times B'$ sind also

Komplement $\overline{R} = \{(x, y) \in (A \times B) \mid (x, y) \notin R\}$

Vereinigung $R \cup S =$
 $\{(x, y) \in (A \cup A') \times (B \cup B') \mid (x, y) \in R \vee (x, y) \in S\}$

Schnitt $R \cap S =$
 $\{(x, y) \in (A \cup A') \times (B \cup B') \mid (x, y) \in R \wedge (x, y) \in S\}$

Differenz $R \setminus S =$
 $\{(x, y) \in (A \cup A') \times (B \cup B') \mid (x, y) \in R \wedge (x, y) \notin S\}$

symmetrische Differenz $R \Delta S = (R \setminus S) \cup (S \setminus R)$

Verkettung von Relationen

Verkettung $R \circ S \subseteq A \times C$

der Relationen $R \subseteq A \times B$ und $S \subseteq B \times C$:

$$R \circ S = \{(x, y) \in A \times C \mid \exists z \in B ((x, z) \in R \wedge (z, y) \in S)\}$$

Beispiele:

- ▶ für $A = \{1, 2, 3\}$, $B = \{a, b, c\}$, $C = \{\diamond, \heartsuit\}$ und
 $R = \{(1, a), (1, c), (2, b), (3, c)\}$, $S = \{(a, \heartsuit), (a, \diamond), (b, \diamond)\}$
 $R \circ S = \{(1, a), (1, c), (2, b), (3, c)\} \circ \{(a, \heartsuit), (a, \diamond), (b, \diamond)\}$
 $= \{(1, \heartsuit), (1, \diamond), (2, \diamond)\}$
- ▶ $\text{istElternteilVon} \circ \text{istElternteilVon} = \text{istGroßelternteilVon}$
- ▶ Für $\leq \subseteq \mathbb{N}^2$ (auch $\mathbb{Z}^2, \mathbb{Q}^2, \mathbb{R}^2$) gilt $(\leq \circ \leq) = \leq$
- ▶ Für $< \subseteq \mathbb{N}^2$ gilt $(< \circ <) \neq <$,
(Gegenbeispiel $(1, 2) \in (< \setminus (< \circ <))$)

Achtung:

Bedeutung von $\circ$ ist abhängig vom Kontext (Relationen, Sprachen)

Eigenschaften der Verkettung von Relationen

WH: $\forall R \subseteq A \times B$ und $S \subseteq B \times C$:

$$R \circ S = \{(x, y) \in A \times C \mid \exists z \in B ((x, z) \in R \wedge (z, y) \in S)\}$$

Für alle Relationen $R \subseteq A \times B, S \subseteq B \times C, T \subseteq C \times D$ gilt:

► $(R \circ S) \circ T = R \circ (S \circ T)$ (assoziativ)

► i.A. $R \circ S \neq S \circ R$ (nicht kommutativ)

Gegenbeispiel: für $R = \{(a, b)\}, S = \{(b, c)\}$
gilt $R \circ S = \{(a, c)\}$, aber $S \circ R = \emptyset$

► $(R \circ S)^{-1} = S^{-1} \circ R^{-1}$ (ÜA)

Verkettung von Relationen ist assoziativ (Beweis)

WH: $\forall R \subseteq A \times B$ und $S \subseteq B \times C$:

$$R \circ S = \{(x, y) \in A \times C \mid \exists z \in B((x, z) \in R \wedge (z, y) \in S)\}$$

zu zeigen:

$$\forall R \subseteq A \times B, S \subseteq B \times C, T \subseteq C \times D : (R \circ S) \circ T = R \circ (S \circ T)$$

Beweis: $\forall (x, y) \in A \times D$:

$$(x, y) \in (R \circ S) \circ T$$

$$\text{gdw. } \exists z \in C : (x, z) \in (R \circ S) \wedge (z, y) \in T$$

$$\text{gdw. } \exists (z, z') \in C \times B : ((x, z') \in R \wedge (z', z) \in S) \wedge (z, y) \in T$$

$$\text{gdw. } \exists (z', z) \in B \times C : ((x, z') \in R \wedge (z', z) \in S) \wedge (z, y) \in T$$

$$\text{gdw. } \exists z' \in B : (x, z') \in R \wedge (\exists z \in C : (z', z) \in S \wedge (z, y) \in T)$$

$$\text{gdw. } \exists z' \in B : (x, z') \in R \wedge (z', y) \in (S \circ T)$$

$$\text{gdw. } (x, y) \in R \circ (S \circ T)$$

Modellierungsbeispiel

Mengen:

$T = \{ \text{Blumen, Pferde, Schiffe} \}$ (Titel des Buches)

$P = \{ \text{Anton, Max, Ilse} \}$ (Personen)

$O = \{ \text{Leipzig, Hamburg, Genf} \}$ (Orte)

$L = \{ \text{D, S} \}$ (Länder)

Relationen:

$\text{hatAutor} \subseteq T \times P$ mit

$\text{hatAutor} = \{ (\text{Blumen, Max}), (\text{Schiffe, Anton}),$
 $(\text{Schiffe, Ilse}), (\text{Pferde, Max}), (\text{Pferde, Ilse}) \}$

$\text{wohntIn} \subseteq P \times O$ mit

$\text{wohntIn} = \{ (\text{Anton, Hamburg}), (\text{Ilse, Leipzig}), (\text{Max, Genf}) \}$

$\text{liegtIn} \subseteq O \times L$ mit

$\text{liegtIn} = \{ (\text{Leipzig, D}), (\text{Hamburg, D}), (\text{Genf, S}) \}$

Verkettungen der Relationen:

► $\text{wohntIn} \circ \text{liegtIn} \subseteq P \times L$ mit

$\text{wohntIn} \circ \text{liegtIn} = \{ (\text{Anton, D}), (\text{Ilse, D}), (\text{Max, S}) \}$

► $\text{hatAutor} \circ \text{wohntIn} \subseteq T \times O$ mit $\text{hatAutor} \circ \text{wohntIn} = \dots$

Modellierungsbeispiel

- ▶ Worüber wird in welchen Ländern geschrieben?

$$R = (\text{hatAutor} \circ \text{wohntIn} \circ \text{liegtIn}) \subseteq T \times L$$

mit $R = \{(\text{Blumen}, S), (\text{Schiffe}, D), (\text{Pferde}, S), (\text{Pferde}, D)\}$

- ▶ In welchen Orten wohnen Autoren des Buches über Pferde?

$$M = \pi_2 \left((\text{hatAutor} \circ \text{wohntIn})|_{\{\text{Pferde}\}} \right) \subseteq O$$

mit $M = \{\text{Genf}, \text{Leipzig}\}$

- ▶ Bücher mit welchen Titeln schreiben Autoren aus Deutschland?

$$M' = \pi_2 \left((\text{hatAutor} \circ \text{wohntIn} \circ \text{liegtIn})^{-1}|_{\{D\}} \right) \subseteq T$$

mit $M' = \{\text{Schiffe}, \text{Pferde}\}$

- ▶ Welche Autoren haben gemeinsam ein Buch geschrieben?

$$R' = (\text{hatAutor}^{-1} \circ \text{hatAutor}) \setminus I_P \subseteq P \times P$$

Was bisher geschah

Modellierung von

- ▶ **Aussagen** in klassischer Aussagenlogik
- ▶ **Daten** (Mengen von Individuen) durch
 - ▶ Mengen
 - ▶ Folgen
 - ▶ Sprachen
- ▶ **Zusammenhängen** und **Eigenschaften** durch Relationen
 - ▶ Definition, Beispiele
 - ▶ Darstellungsformen
 - ▶ Operationen auf Relationen: Mengenoperationen, inverse Relation, Verkettung, Projektionen, Einschränkung,
 - ▶ Modellierungsbeispiele

Komponentenweises Produkt von Relationen

Das **komponentenweise Produkt** (auch direktes Produkt)

der Relationen $R \subseteq A^2$ und $S \subseteq B^2$

ist die Relation $(R \cdot_k S) \subseteq (A \times B)^2$ mit $\forall ((x, y), (x', y')) \in (A \times B)^2 :$

$$((x, y), (x', y')) \in (R \cdot_k S) \text{ gdw. } (((x, x') \in R) \wedge ((y, y') \in S))$$

Beispiele:

- ▶ $A = \{a, b, c\}$, $B = \{0, 1\}$,
 $R \subseteq A^2$ mit $R = \{(a, c), (b, c)\}$, $S \subseteq B^2$ mit $S = \{(0, 0), (1, 0)\}$
 $(R \cdot_k S) =$
 $\{((a, 0), (c, 0)), ((a, 1), (c, 0)), ((b, 0), (c, 0)), ((b, 1), (c, 0))\}$
- ▶ Münzspiel mit zwei Spielern A und B :
 $R \subseteq (\mathbb{N}^*)^2$ Übergänge nach Spielregeln für Ein-Personen-Spiel
 $S = \{(A, B), (B, A)\} \subseteq \{A, B\}^2$ Spieler A und B ziehen abwechselnd
 $(R \cdot_k S) \subseteq (\mathbb{N}^* \times \{A, B\})^2$ Übergänge im Zwei-Personen-Spiel

Lexikographisches Produkt von Relationen

Das **lexikographische Produkt**

der Relationen $R \subseteq A^2$ und $S \subseteq B^2$

ist die Relation $(R \cdot_I S) \subseteq (A \times B)^2$ mit

$\forall ((x, y), (x', y')) \in (A \times B) \times (A \times B) :$

$((x, y), (x', y')) \in (R \cdot_I S)$ gdw. $((x, x') \in R) \vee ((x = x') \wedge ((y, y') \in S))$

Beispiele:

- ▶ $R \subseteq \{a, b\}^2$, $S \subseteq \{0, 1, 2\}^2$,
mit $R = \{(a, b)\}$ und $S = \{(0, 1), (0, 2), (1, 2)\}$

$$(R \cdot_I S) = \left\{ \begin{array}{l} (a0, b0), (a0, b1), (a0, b2), (a1, b0), (a1, b1), (a1, b2), \\ (a2, b0), (a2, b1), (a2, b2), \\ (a0, a1), (a0, a2), (a1, a2), (b0, b1), (b0, b2), (b1, b2) \end{array} \right\}$$

- ▶ Skat mit Trumpf $\heartsuit$: $R = \{(\heartsuit, \diamondsuit), (\heartsuit, \spadesuit), (\heartsuit, \clubsuit)\} \subseteq \{\diamondsuit, \heartsuit, \spadesuit, \clubsuit\}^2$
 $S \subseteq \{A, K, D, B, 10, 9, 8, 7\}^2$ mit Wert nach Reihenfolge
 $(R \cdot_I S)$ höhere Karte: Trumpf oder gleiche Farbe und größerer Wert
- ▶ Sortieren zuerst nach Gruppe, dann nach Familienname

Zweistellige Relationen auf einer Menge

Spezialfall:

zweistellige Relation auf einer Menge M : $R \subseteq M \times M$

Verkettung der Relationen $R \subseteq M \times M$ und $S \subseteq M \times M$:

$$R \circ S = \{(x, y) \mid \exists z \in M : (x, z) \in R \wedge (z, y) \in S\}$$

Beispiel:

$$M = \{a, b, c\}$$

$$R = \{(a, a), (b, c)\}$$

$$S = \{(a, c), (c, b)\}$$

$$R \circ S = \{(a, c), (b, b)\}$$

$$S \circ R = \{(c, c)\}$$

Darstellung als Diagramm

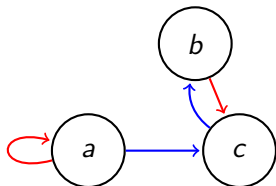
Knotenmenge M , Relation $R \subseteq M^2$

Kante von $x \in M$ nach $y \in M$ gdw. $(x, y) \in R$

$$M = \{a, b, c\}$$

$$R = \{(a, a), (b, c)\}$$

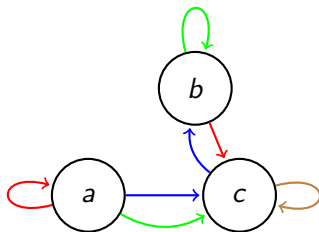
$$S = \{(a, c), (c, b)\}$$



Verkettung als Pfade mit passender Markierung

$$R \circ S = \{(a, c), (b, b)\}$$

$$S \circ R = \{(c, c)\}$$



Darstellung als Matrix

mit Booleschen Einträgen

für $M = \{a, b, c\}$, $R \subseteq M^2$ und $S \subseteq M^2$

$$R = \{(a, a), (b, c)\} \quad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

$$S = \{(a, c), (c, b)\} \quad \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

Verkettung als Matrixmultiplikation mit Booleschen Operationen

$$R \circ S = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$S \circ R = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Eigenschaften binärer Relationen

Eine binäre Relation $R \subseteq M^2$ heißt

reflexiv gdw. $\forall x \in M : ((x, x) \in R)$ $(I_M \subseteq R)$
z.B. $R = \{(a, a), (a, b), (b, b)\} \subseteq \{a, b\}^2$, $=$, $\leq$, $\sqsubseteq$, $\subseteq$

irreflexiv gdw. $\forall x \in M : ((x, x) \notin R)$ $(R \cap I_M = \emptyset)$
z.B. $R = \{(a, b), (b, a)\} \subseteq \{a, b\}^2$,
 $<$, $\subset$, istGeschwisterVon, istKindVon

symmetrisch gdw. $\forall x, y \in M : ((x, y) \in R \rightarrow (y, x) \in R)$ $(R^{-1} \subseteq R)$
z.B. $R = \{(a, b), (b, a), (b, b)\} \subseteq \{a, b\}^2$,
 $=$, istGeschwisterVon

asymmetrisch gdw. $\forall x, y \in M : ((x, y) \in R \rightarrow (y, x) \notin R)$ $(R \cap R^{-1} = \emptyset)$
z.B. $R = \{(a, b)\} \subseteq \{a, b\}^2$, $<$, istKindVon,

antisymmetrisch gdw. $\forall x, y \in M : (((x, y) \in R \wedge (y, x) \in R) \rightarrow x = y)$ $(R \cap R^{-1} \subseteq I_M)$
z.B. $R = \{(a, b), (b, b)\} \subseteq \{a, b\}^2$, $\leq$, $\sqsubseteq$, $\subseteq$,

transitiv gdw. $\forall x, y, z \in M : (((x, y) \in R \wedge (y, z) \in R) \rightarrow (x, z) \in R)$ $(R \circ R \subseteq R)$

z.B. $R = \{(c, b), (a, b), (a, c)\} \subseteq \{a, b, c\}^2$,
 $=$, $\leq$, $<$, $\sqsubseteq$, $\subseteq$, istVorfahreVon

Quasiordnungen

Relation $R \subseteq M^2$ heißt **Quasiordnung** gdw. R reflexiv und transitiv

Beispiele:

- ▶ $R \subseteq \{a, b, c\}^2$ mit
 $R = \{(a, a), (a, b), (b, b), (b, c), (a, c), (c, c)\}$
- ▶ $T \subseteq (\text{Menge aller Personen})^2$ mit
 $(p, q) \in T$ gdw. p höchstens so alt ist wie q
- ▶ $S \subseteq \mathbb{Z}^2$ mit $S = \{(a, b) \in \mathbb{Z}^2 \mid |a| \geq |b|\}$

Relation $R \subseteq M^2$ heißt **Äquivalenzrelation**
gdw. R reflexiv, transitiv und symmetrisch
(symmetrische Quasiordnung)

Relation $R \subseteq M^2$ heißt **Halbordnung**
gdw. R reflexiv, transitiv und antisymmetrisch
(antisymmetrische Quasiordnung)

Totale (= lineare) Ordnungen

Jede Halbordnung $R \subseteq M^2$ mit

$$\forall (a, b) \in M^2 \quad ((a, b) \in R \vee (b, a) \in R) \quad (\text{alternativ})$$

heißt **totale** (lineare) Ordnung.

(alle Elemente miteinander vergleichbar)

Beispiele:

- ▶ $R \subseteq (\{a, b, c\})^2$ mit
 $R = \{(a, a), (a, b), (a, c), (c, b), (b, b), (c, c)\}$
- ▶ $\leq \subseteq \mathbb{N}^2$ ($\leq \subseteq \mathbb{Z}^2$, $\leq \subseteq \mathbb{R}^2$)
- ▶ $R \subseteq (\text{Menge aller MIB-Studenten})^2$ mit
 $(p, q) \in R$ gdw. (Matrikelnr. von p) $\leq$ (Matrikelnr. von q)
- ▶ alphabetische Ordnung der Wörter im Wörterbuch
- ▶ Halbordnungen, aber keine totalen Ordnungen sind
z.B. $\sqsubseteq \subseteq (\{a, b\}^*)^2$, $| \subseteq \mathbb{N}^2$ (ÜA), $\subseteq \subseteq 2^{\{a, b\}}$

Strikte Ordnungen

Jede irreflexive, transitive und asymmetrische Relation $R \subseteq M^2$ heißt **strikte Ordnung**.

(Strikte Ordnungen sind i.A. keine Halbordnungen.)

Beispiele:

- ▶ $R \subseteq (\{a, b, c\})^2$ mit $R = \{(a, c), (b, c)\}$
- ▶ $< \subseteq \mathbb{N}$ ($< \subseteq \mathbb{Z}$, $< \subseteq \mathbb{R}$)
- ▶ alphabetische Ordnung der Wörter im Wörterbuch

Für jede Halbordnung $R \subseteq M^2$ heißt $R \setminus I_M$ **strikter Teil** von R .

Beispiele:

- ▶ $<$ ist strikter Teil von $\leq$
- ▶ $\subset$ ist strikter Teil von $\subseteq$

Was bisher geschah

Modellierung von

- ▶ **Aussagen** durch Logiken
- ▶ **Daten** durch
 - ▶ Mengen, Folgen, Sprachen
Darstellungsformen, Anwendungen, Beziehungen, Operationen
- ▶ **Zusammenhängen** und **Eigenschaften** durch **Relationen**
Darstellungsformen, Anwendungen, Beziehungen, Operationen
 - ▶ Eigenschaften binärer Relationen $R \subseteq M^2$
 - ▶ spezielle binäre Relationen:
 - ▶ Quasiordnungen
 - ▶ Äquivalenzrelationen
 - ▶ Halbordnungen
 - ▶ totale Ordnungen
 - ▶ strikte Ordnungen

Hüllen binärer Relationen

Die reflexive (symmetrische, transitive, reflexiv-transitive) **Hülle** einer Relation $R \subseteq M^2$ ist die kleinste (bzgl. $\subseteq$) Relation $S \subseteq M^2$ mit

1. $R \subseteq S$ und
2. S reflexiv (symmetrisch, transitiv, reflexiv und transitiv).

reflexive Hülle von R : $R \cup I_M$

symmetrische Hülle von R : $R \cup R^{-1}$

Wiederholung: Verkettung $\circ$ der Relationen $R \subseteq M^2$ und $S \subseteq M^2$

$$R \circ S = \{(x, z) \in M^2 \mid \exists y \in M : (x, y) \in R \wedge (y, z) \in S\}$$

Iterierte Verkettung von $R \subseteq M^2$ mit sich selbst:

$$\begin{aligned} R^0 &= I_M \\ R^{n+1} &= R^n \circ R \\ R^+ &= \bigcup_{n \in \mathbb{N} \setminus \{0\}} R^n \subseteq M^2 \quad \text{transitive Hülle} \end{aligned}$$

$$R^* = \bigcup_{n \in \mathbb{N}} R^n \subseteq M^2 \quad \text{reflexiv-transitive Hülle}$$

Beispiel

$$R \subseteq (\{a, b, c\})^2 \quad \text{mit} \quad R = \{(a, b), (b, c)\}$$

$$R^{-1} = \{(b, a), (c, b)\}$$

$$R \cup R^{-1} = \{(a, b), (b, c), (b, a), (c, b)\} \quad (\text{symmetrische Hülle})$$

$$R^0 = \{(a, a), (b, b), (c, c)\}$$

$$R \cup R^0 = \{(a, b), (b, c), (a, a), (b, b), (c, c)\} \quad (\text{reflexive Hülle})$$

$$R^2 = \{(a, c)\}$$

$$R^3 = \emptyset \quad (= R^n \text{ für beliebige } n \geq 3)$$

$$R^+ = R \cup R^2 \cup R^3 \cup \dots = \{(a, b), (b, c), (a, c)\} \\ (\text{transitive Hülle})$$

$$R^* = R^0 \cup R^+ = \{(a, a), (b, b), (c, c), (a, b), (b, c), (a, c)\} \\ (\text{reflexiv-transitive Hülle})$$

Modellierungsbeispiel Münzspiel

- Menge der Spielzustände (eingeschränkte Spielvariante))

$$Q \subset \left\{ (x_1, x_2, \dots, x_5) \in \{0, 1, \dots, 4\}^5 \mid \sum_{i=1}^5 x_i = 4 \right\}$$

- Startzustand $00400 \in Q$,
- Übergangsrelation $T \subseteq Q^2$ mit

$$T = \left\{ \begin{array}{lll} (00400, 01210) & , & (01210, 02020) & , & (02020, 10120) \\ (02020, 02101) & , & (10120, 10201) & , & (02101, 10201), \\ (10201, 11011) \end{array} \right\}$$

Hüllen der Übergangsrelation enthalten
zusätzliche Möglichkeiten für Spielzüge:

- reflexive Hülle $T \cup I_Q$ (mit Aussetzen)
- symmetrische Hülle $T \cup T^{-1}$ (mit umgekehrten Züge)
- transitive Hülle T^+
(mehrere Züge auf einmal, aber wenigstens einer)
- reflexiv-transitive Hülle T^*
(beliebig viele Züge auf einmal, Aussetzen erlaubt)

Äquivalenzrelationen

Relation $R \subseteq M^2$ heißt **Äquivalenzrelation**
gdw. R reflexiv, transitiv und symmetrisch
(symmetrische Quasiordnung)

Beispiele:

- ▶ $R \subseteq \{a, b, c\}^2$ mit $R = \{(a, a), (a, b), (b, a), (b, b), (c, c)\}$
- ▶ Relation „haben dieselbe Farbe“ für Paare von Skatkarten
 $R \subseteq (F \times W)^2$ für $F = \{\diamond, \heartsuit, \spadesuit, \clubsuit\}$ und
 $W = (\{7, 8, 9, 10\} \cup \{B, D, K, A\})$ mit
 $R = \{((f, w_1), (f, w_2)) \mid (f \in F) \wedge (w_1 \in W) \wedge (w_2 \in W)\}$
- ▶ Relation „sind im selben Studiengang“ für Studenten
- ▶ $R \subseteq (A^*)^2$ mit $R = \{(u, v) \in (A^*)^2 \mid |u| = |v|\}$ (gleiche Länge)
- ▶ Gleichmächtigkeit von Mengen: $(A, B) \in R$ gdw. $|A| = |B|$
- ▶ Semantische Äquivalenz aussagenlogischer Formeln $\equiv \subseteq \text{AL}(P)$
- ▶ Äquivalenz regulärer Ausdrücke
- ▶ Ähnlichkeit von Dreiecken
- ▶ für jede Menge M : $I_M \subseteq M^2$ mit $I_M = \{(x, x) \mid x \in M\}$

Äquivalenzklassen

Zu jeder Äquivalenzrelation $R \subseteq M^2$ heißen

- ▶ zwei Elemente $a \in M$ und $b \in M$ genau dann **äquivalent bzgl. R** , wenn $(a, b) \in R$
- ▶ die Menge aller zu $a \in M$ äquivalenten Elemente aus M

$$[a]_R = \{x \in M \mid (a, x) \in R\}$$

Äquivalenzklasse von a bzgl. R

Beispiele:

- ▶ Äquivalenzklassen bzgl. der Äquivalenzrelation $R \subseteq \{a, b, c\}^2$ mit $R = \{(a, a), (a, b), (b, a), (b, b), (c, c)\}$:
 $[a]_R = \{a, b\} = [b]_R$ und $[c]_R = \{c\}$
- ▶ Äquivalenzklassen bzgl. der Äquivalenzrelation $G \subseteq (F \times W)^2$ mit
 $G = \{((f, w), (f', w')) \in (F \times W)^2 \mid f = f'\}$
„haben dieselbe Farbe“ auf Skatkarten:
 $K_{\diamond}, K_{\heartsuit}, K_{\spadesuit}, K_{\clubsuit}$ mit z.B.
 $K_{\diamond} = [(\diamond, 9)]_G = \{(\diamond, w) \mid w \in \{7, 8, 9, 10, B, D, K, A\}\}$

Äquivalenzrelationen und disjunkte Zerlegungen

1. Jede Äquivalenzrelation $R \subseteq M^2$ definiert eine disjunkte Zerlegung von M in die Menge aller Äquivalenzklassen

$$\{[a]_R \mid a \in M\}$$

2. Jede (disjunkte) Zerlegung $P = \{M_i \mid i \in I\}$ einer Menge M definiert eine Äquivalenzrelation

$$R_P \subseteq M^2 \quad \text{mit} \quad R_P = \{(a, b) \mid \exists i \in I : \{a, b\} \subseteq M_i\}$$

Beispiele:

- ▶ Relation $R \subseteq \{a, b, c\}^2$ mit $R = \{(a, a), (a, b), (b, a), (b, b), (c, c)\}$ definiert Zerlegung $\{[a]_R, [b]_R, [c]_R\} = \{\{a, b\}, \{c\}\}$ der Menge $\{a, b, c\}$
- ▶ Zerlegung $F = \{K_\diamond, K_\heartsuit, K_\spadesuit, K_\clubsuit\}$ der Menge aller Skatkarten definiert die Relation R_F (haben dieselbe Farbe)
- ▶ Zerlegung $P = \{3\mathbb{N}, 3\mathbb{N} + 1, 3\mathbb{N} + 2\}$ definiert die Relation $R_P = \{(m, n) \in \mathbb{N}^2 \mid 3 \mid (m - n)\}$

Halbordnungen (partielle Ordnungen)

Relation $R \subseteq M^2$ heißt **Halbordnung**
gdw. R reflexiv, transitiv und antisymmetrisch
(antisymmetrische Quasiordnung)

$a \in M$ und $b \in M$ heißen genau dann **vergleichbar** bzgl. R ,
wenn $(a, b) \in R$ oder $(b, a) \in R$

Beispiele:

- ▶ $R \subseteq \{a, b, c, d\}^2$ mit
 $R = \{(a, a), (a, b), (a, c), (a, d), (b, b), (b, d), (c, c), (d, d)\}$
- ▶ $\leq \subseteq \mathbb{N}^2$ ($\leq \subseteq \mathbb{Z}^2$, $\leq \subseteq \mathbb{R}^2$)
- ▶ $R \subseteq (\text{Menge aller MIB-Studenten})^2$ mit
 $(p, q) \in R$ gdw. (Matrikelnr. von p) $\leq$ (Matrikelnr. von q)
- ▶ $\sqsubseteq \subseteq (\{a, b\}^*)^2$
- ▶ Teilerrelation $| \subseteq \mathbb{N}^2$ mit $a|b$ gdw. $\exists c \in \mathbb{Z} \mid ac = b$
(Warum ist $| \subseteq \mathbb{Z}^2$ keine Halbordnung?)
- ▶ Teilmengenrelation $\subseteq \subseteq (2^{\{a, b, c\}})^2$

Hasse-Diagramme von Halbordnungen

WH: $R \subseteq M^2$ heißt Halbordnung gdw. R reflexiv, transitiv, antisymmetrisch

graphische Darstellung von Halbordnungen auf endlichen Mengen mit folgenden Konventionen:

- ▶ Pfeilspitzen weglassen, stattdessen optische Ausrichtung (vertikal, größere Werte oben)
- ▶ reflexive Kanten (Schlingen) weglassen
- ▶ transitive Kanten weglassen

Beispiele (Tafel):

- ▶ Teiler-Halbordnung $| \subseteq \{0, \dots, 6\}^2$
- ▶ $\leq \subseteq \{0, \dots, 6\}^2$
- ▶ $\subseteq \subseteq (2^{\{a,b,c\}})^2$

Minimale Elemente in HO

Für Halbordnung $R \subseteq M^2$:

x heißt **minimal** bzgl. R in M gdw.

1. $x \in M$ und
2. $\forall y \in M : ((y, x) \in R \rightarrow y = x)$

Enthält M **genau ein** bzgl. R minimales Element,
dann heißt dieses **Minimum** bzgl. R in M .

Beispiele:

- ▶ bzgl. $\leq$ ist 2 minimal in $\{2, 3, 6\}$,
2 ist auch Minimum bzgl. $\leq$ in $\{2, 3, 6\}$,
- ▶ 3 ist bzgl. $\leq$ nicht minimal in $\{2, 3, 6\}$,
- ▶ 0 ist Minimum von $\mathbb{N}$ bzgl. $\leq$
- ▶ $\mathbb{Z}$ enthält kein minimales Element bzgl. $\leq$
- ▶ bzgl. $|$ sind 2 und 3 minimal in $\{2, 3, 6\}$,
aber in $\{2, 3, 6\}$ existiert kein Minimum bzgl. $|$

Maximale Elemente in HO

Für Halbordnung $R \subseteq M^2$:

x heißt **maximal** bzgl. R in M gdw.

1. $x \in M$ und
2. $\forall y \in M : ((x, y) \in R \rightarrow y = x)$

Enthält M **genau ein** bzgl. R maximales Element,
dann heißt dieses **Maximum** bzgl. R in M .

Beispiele:

- ▶ bzgl. $\leq$ ist 6 maximal in $\{2, 3, 6\}$,
6 ist auch Maximum bzgl. $\leq$ in $\{2, 3, 6\}$,
- ▶ 3 ist bzgl. $\leq$ nicht maximal in $\{2, 3, 6\}$,
- ▶ bzgl. $|$ ist 6 maximal in $\{2, 3, 6\}$,
6 ist auch Maximum bzgl. $|$ in $\{2, 3, 6\}$,
- ▶ bzgl. $\sqsubseteq$ sind b, aa und aba maximal in $\{a, b, aa, aba\}$,
aber in $\{a, b, aa, aba\}$, existiert kein Maximum bzgl. $\sqsubseteq$

Was bisher geschah

Modellierung von

- ▶ **Aussagen** durch Logiken
- ▶ **Daten** durch
 - ▶ Mengen, Folgen, Sprachen
 - Darstellungsformen, Anwendungen, Beziehungen, Operationen
- ▶ **Zusammenhängen** und **Eigenschaften** durch **Relationen**
 - Darstellungsformen, Anwendungen, Beziehungen, Operationen
 - ▶ Eigenschaften binärer Relationen $R \subseteq M^2$
 - ▶ reflexive, symmetrische, transitive Hüllen
 - ▶ spezielle binäre Relationen:
 - ▶ Äquivalenzrelationen, Zerlegung in Äquivalenzklassen
 - ▶ Halbordnungen, Hasse-Diagramm, minimale / maximale Elemente
 - ▶ totale Ordnungen
 - ▶ strikte Ordnungen

WH (aus Mathematik): Funktionen

Funktion $f : A \rightarrow B$: spezielle Relation $f \subseteq A \times B$

Relation $f \subseteq A \times B$ heißt genau dann **partielle Funktion**,
wenn $\forall b, c \in B : ((a, b) \in f \wedge (a, c) \in f) \rightarrow b = c$ (rechtseindeutig)
(also wenn $\forall a \in A : |\pi_2(f|_{\{a\}})| \leq 1$)

Relation $f \subseteq A \times B$ heißt genau dann (totale) **Funktion**,
wenn f partielle Funktion und $\forall a \in A \exists b \in B : (a, b) \in f$ (linkstotal)
(also wenn $\forall a \in A : |\pi_2(f|_{\{a\}})| = 1$)

Beispiele:

- ▶ Relation zwischen Personen und ihrem Geburtstag ist eine totale Funktion
- ▶ Relation zwischen Tagen und Personen, die an diesem Tag Geburtstag haben, ist keine Funktion
- ▶ $R \subseteq \mathbb{Z}^2$ mit $R = \{(x, x^2) \mid x \in \mathbb{Z}\}$ ist (totale) Funktion
- ▶ $R^{-1} \subseteq \mathbb{Z}^2$ mit $R^{-1} = \{(x^2, x) \mid x \in \mathbb{Z}\}$ ist keine Funktion
- ▶ $S \subseteq \mathbb{N}^2$ mit $S = \{(x^2, x) \mid x \in \mathbb{N}\}$ ist partielle Funktion
- ▶ $T \subseteq \mathbb{R}^2$ mit $T = \{(\sin x, x) \mid x \in \mathbb{R}\}$ ist keine Funktion
- ▶ für jede Menge A ist die Identitätsrelation $I_A \subseteq A^2$ mit $I_A = \{(x, x) \mid x \in A\}$ eine Funktion (identische Funktion auf A)

Definition von Funktionen

Funktion $f : A \rightarrow B$ wird definiert durch Angabe von:

Typ (Signatur), bestehend aus (Syntax)

Definitionsbereich : Menge A

Wertebereich : Menge B

Werte Zuordnung f (Semantik)

extensional z.B. $f = \{a \mapsto 1, b \mapsto 0\}$
(statt $f = \{(a, 1), (b, 0)\}$)

intensional z.B. $f : \mathbb{N} \rightarrow \mathbb{N}$ mit $\forall x \in \mathbb{N} : f(x) = x^2$

Graph von f : Relation $f \subseteq A \times B = \{(x, f(x)) \mid x \in A\}$

Auf jeder Menge A ist die Identitätsrelation $I_A \subseteq A^2$
der Graph der **identischen** Funktion (Identität) auf A

$$1_A : A \rightarrow A \quad \text{mit} \quad \forall x \in A : 1_A(x) = x$$

Mehrstellige Funktionen auf einer Menge

Funktion $f : A \rightarrow B$ mit

$A = B^n$: Funktion $f : B^n \rightarrow B$ heißt auch
 n -stellige Funktion auf B

Beispiel: $f : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ mit

$$\forall (x, y) \in \mathbb{N}^2 : f(x, y) = 2x + y$$

$A = B^0 = \{\varepsilon\}$:

nullstellige Funktionen $f : B^0 \rightarrow B$ ($f : \{\varepsilon\} \rightarrow B$)
heißen auch **Konstanten** $f \in B$.

Beispiel: $c : \mathbb{N}^0 \rightarrow \mathbb{N}$ mit $c = 3$

Achtung:

n -stellige Funktionen $f : B^n \rightarrow B$ sind

$n + 1$ -stellige Relationen $f \subseteq B^{n+1}$ auf B

Abgeleitete Funktionen

Zu jeder Funktion $f : A \rightarrow B$ sind definiert

	$f(A)$	$= \pi_2(f) \subseteq B$	Bild von f ,
für $M \subseteq A$:	$f(M)$	$= \pi_2(f _M) \subseteq B$	Bild von M unter f ,
für $a \in A$:	$f(a)$	$= \pi_2(f _{\{a\}}) \in B$	Bild von a unter f ,
für $N \subseteq B$	$f^{-1}(N)$	$= \pi_2(f^{-1} _N) \subseteq A$	Urbild von N unter f ,
für $b \in B$	$f^{-1}(b)$	$= \pi_2(f^{-1} _{\{b\}}) \subseteq A$	Urbild von b unter f

Beispiel: Für $A = \{a, b, c\}$, $B = \{1, 2, 3\}$ und

$f : \{a, b, c\} \rightarrow \{1, 2, 3\}$ mit $f(a) = f(c) = 2$, $f(b) = 1$ gilt

- ▶ $f(A) = \{1, 2\} \subseteq B$,
- ▶ $f(\{a, c\}) = \{2\} \subseteq B$, $f(\{b, c\}) = \{1, 2\} \subseteq B$,
- ▶ $f^{-1}(\{1, 3\}) = \{b\} \subseteq A$,
- ▶ $f^{-1}(\{2\}) = f^{-1}(2) = \{a, c\} \subseteq A$ und $f^{-1}(3) = \emptyset \subseteq A$

Für jede (totale) Funktion $f : A \rightarrow B$ gilt $f^{-1}(B) = A$
(aber nicht notwendig $f(A) = B$)

WH: Eigenschaften von Funktionen

Funktion $f : A \rightarrow B$ heißt

- injektiv** , falls für jedes $b \in B$ gilt $|f^{-1}(b)| \leq 1$,
- surjektiv** , falls für jedes $b \in B$ gilt $|f^{-1}(b)| \geq 1$,
- bijektiv** , falls für jedes $b \in B$ gilt $|f^{-1}(b)| = 1$,

Eine Funktion ist genau dann bijektiv,
wenn sie injektiv und surjektiv ist.

Beispiele:

- ▶ $f : \mathbb{N} \rightarrow \mathbb{N}$ mit $\forall x \in \mathbb{N} : f(x) = x^2$ ist injektiv, nicht surjektiv
- ▶ $f : \mathbb{Z} \rightarrow \mathbb{N}$ mit $\forall x \in \mathbb{Z} : f(x) = |x|$ ist surjektiv, nicht injektiv
- ▶ $f : \mathbb{Z} \rightarrow \mathbb{Z}$ mit $\forall x \in \mathbb{Z} : f(x) = x^2$ ist
weder injektiv noch surjektiv
- ▶ $f : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ mit $\forall x \in \mathbb{R}_{\geq 0} : f(x) = x^2$ ist
injektiv und surjektiv

Mengen von Funktionen

Menge aller (totalen) Funktionen von A nach B :

$$B^A = \{f : A \rightarrow B\} = \{f \subseteq A \times B \mid \forall a \in A : |\pi_2(f|_{\{a\}})| = 1\}$$

Beispiel: $\{0, 1\}^{\{p, q, r\}} = \{W : \{p, q, r\} \rightarrow \{0, 1\}\}$

Für alle endlichen Mengen A, B gilt $|B^A| = |B|^{|A|}$.

(Es existieren $|B|^{|A|}$ verschiedene Funktionen von A nach B .)

Beispiele:

- ▶ $|\{0, 1\}^{\{p, q, r\}}| = |\{0, 1\}|^{|\{p, q, r\}|} = 2^3 = 8$ Funktionen
 $W : \{p, q, r\} \rightarrow \{0, 1\}$
- ▶ $3^{|A|}$ Funktionen $f : A \rightarrow \{\text{rot, grün, blau}\}$
- ▶ $|\{1, 2, 3, 4\}|^{|A|} = 4^{|A|}$ Funktionen $f : A \rightarrow \{1, 2, 3, 4\}$
- ▶ $|(\{0, 1\}^2)^A| = (|\{0, 1\}^2|)^{|A|} = 4^{|A|}$ Funktionen $f : A \rightarrow \{0, 1\}^2$
- ▶ $|A^{(\{0, 1\}^2)}| = |A|^{(|\{0, 1\}^2|)} = |A|^4$ Funktionen $f : \{0, 1\}^2 \rightarrow A$

Funktionen auf Intervallen $\{1, \dots, n\} \subset \mathbb{N}$

Funktion $f : \{1, \dots, n\} \rightarrow B$ ($f \in B^{\{1, \dots, n\}}$, oft kürzer $f \in B^n$)

Beispiel:

$f : \{1, \dots, 5\} \rightarrow \mathbb{N}$, wobei $\forall n \in \{1, 2, \dots, 5\} : f(n) = 7n + 2$

$f(1) = 9, f(2) = 16, f(3) = 23, f(4) = 30, f(5) = 37$

alternative Darstellung $f_1 = 9, f_2 = 16, f_3 = 23, f_4 = 30, f_5 = 37$,
als Folge $f = (9, 16, 23, 30, 37) = (7n + 2)_{i \in \{1, \dots, 5\}}$

Jede **Funktion** $f : \{1, \dots, n\} \rightarrow B$ ($f \in B^{\{1, \dots, n\}}$) repräsentiert
eine eindeutig definierte **Folge** $(f(i))_{i \in \{1, \dots, n\}} \in B^n$

Jede Folge $(a_i)_{i \in \{1, \dots, n\}} \in B^n$ repräsentiert
eine eindeutig definierte Funktion

$a : \{1, \dots, n\} \rightarrow B$ mit $\forall i \in \{1, \dots, n\} : a(i) = a_i$

Analog: Funktionen auf Produkten von Intervallen

$f : (\{1, \dots, m\} \times \{1, \dots, n\}) \rightarrow B$

repräsentieren Matrix $f \in B^{\{1, \dots, m\} \times \{1, \dots, n\}}$, oft kürzer $f \in B^{m \times n}$

analog: mehrdimensionale Matrizen $f \in B^{n_1 \times \dots \times n_k}$

Funktionen nach $\{0, 1\}$

Für endliche Mengen A und B

Anzahl

- ▶ nullstellige Funktionen $f : A^0 \rightarrow \{0, 1\}$ ($f \in \{0, 1\}^{\{\varepsilon\}}$)
 $f_0 = 0, f_1 = 1$ (Konstanten) $2^{|A^0|} = 2^1 = 2$
- ▶ einstellige Funktionen $f : A \rightarrow \{0, 1\}$ ($f \in \{0, 1\}^A$) $2^{|A|}$
Beispiel: für $A = \{a, b\}$

$$f_{00} = \{a \mapsto 0, b \mapsto 0\}$$

$$f_{01} = \{a \mapsto 0, b \mapsto 1\}$$

$$f_{10} = \{a \mapsto 1, b \mapsto 0\}$$

$$f_{11} = \{a \mapsto 1, b \mapsto 1\}$$

mögliche Urbilder der 1:

$$f_{00}^{-1}(1) = \emptyset, f_{01}^{-1}(1) = \{b\}, f_{10}^{-1}(1) = \{a\}, f_{11}^{-1}(1) = \{a, b\}$$

für beliebige Menge A :

$$\{f^{-1}(1) \subseteq A \mid f : A \rightarrow \{0, 1\}\} = \{f^{-1}(1) \mid f \in \{0, 1\}^A\} = 2^A$$

(Potenzmenge von A)

- ▶ zweistellige Funktionen $f : A \times B \rightarrow \{0, 1\}$ $2^{|A||B|}$
- ▶ zweistellige Funktionen $f : A^2 \rightarrow \{0, 1\}$ $2^{|A|^2}$
- ▶ n -stellige Funktionen $f : A^n \rightarrow \{0, 1\}$ $2^{|A|^n}$

Charakteristische Funktion einer Menge

Menge U definiert für jede Teilmenge $A \subseteq U$ die
charakteristische Funktion $\chi_A : U \rightarrow \{0, 1\}$ mit

$$\forall x \in U : \chi_A(x) = \begin{cases} 1 & \text{falls } x \in A \\ 0 & \text{sonst} \end{cases}$$

Beispiel: für $U = \{a, b, c, d\}$ und $A = \{a, d\} \subseteq U$ gilt
 $\chi_A : U \rightarrow \{0, 1\}$ mit $\chi_A = \{a \mapsto 1, b \mapsto 0, c \mapsto 0, d \mapsto 1\}$
(χ_A ordnet jedem $x \in U$ den Wahrheitswert der Aussage $x \in A$ zu.)
Umgekehrt definiert jede Funktion $f : U \rightarrow \{0, 1\}$ eine Teilmenge

$$f^{-1}(1) \subseteq U$$

Für **endliche** Mengen $U = \{a_1, \dots, a_n\}$ und
beliebige feste Anordnung $[a_1, \dots, a_n]$ aller Elemente aus U
lässt sich jede Teilmenge $A \subseteq U$ durch das eindeutige
Binärwort (charakteristischer Vektor) $b_1 \dots b_n \in \{0, 1\}^n$ mit $b_i = \chi_A(a_i)$
repräsentieren.

Multimengen (Vielfachmengen, Bags)

zur Beschreibung von Elementen mit **Vielfachheit**

Beispiele:

- ▶ Bibliothek mit mehreren Exemplaren von Büchern
- ▶ Vorrat an Münzen
- ▶ Rommé-Blatt

Teilmenge A einer Menge U (Universum)

als charakteristische Funktion von A bzgl. U :

$$\chi_A : U \rightarrow \{0, 1\} \quad \text{mit} \quad \forall x \in U : \chi_A(x) = |A \cap \{x\}|$$

Multimenge A über Menge U : Funktion $A : U \rightarrow \mathbb{N}$

alternative Darstellungen von Multimengen:

z.B. für $U = \{a, b, c, d\}$ und $A : U \rightarrow \mathbb{N}$ mit

$A(a) = 3, A(b) = 3, A(c) = 0, A(d) = 1$ durch

- ▶ $\{a \mapsto 3, b \mapsto 3, c \mapsto 0, d \mapsto 1\}$, auch $\{a \mapsto 3, b \mapsto 3, d \mapsto 1\}$
(dann implizit $x \mapsto 0$ für alle nicht enthaltenen $x \in U$)
- ▶ als Relation $A \subseteq U \times (\mathbb{N} \setminus \{0\})$ mit $\{(a, 3), (b, 3), (d, 1)\}$
- ▶ $\{a, b, a, b, d, b, a\}$ (Reihenfolge egal)

Mächtigkeit der Multimenge $A : U \rightarrow \mathbb{N}$: $|A| = \sum_{x \in U} A(x)$

Beziehungen zwischen Multimengen

Multimenge $A : U \rightarrow \mathbb{N}$ ist genau dann **Teil(multi)menge** der Multimenge $B : U \rightarrow \mathbb{N}$, wenn

$$\forall x \in U : A(x) \leq B(x)$$

(A „punktweise“ $\leq B$)

Zwei Multimengen $A : U \rightarrow \mathbb{N}$ und $B : U \rightarrow \mathbb{N}$ sind genau dann **gleich**, wenn

$$\forall x \in U : A(x) = B(x)$$

(Charakteristische Funktionen von) Mengen $A \subseteq U$ sind spezielle Multimengen mit der Eigenschaft $\forall x \in U : A(x) \leq 1$

Operationen auf Multimengen

Für Multimengen $A : U \rightarrow \mathbb{N}$ und $B : U \rightarrow \mathbb{N}$

Vereinigung $\forall x \in U : (A \cup B)(x) = \max(A(x), B(x))$

disjunkte Vereinigung $\forall x \in U : (A \dot{\cup} B)(x) = A(x) + B(x)$

Schnitt $\forall x \in U : (A \cap B)(x) = \min(A(x), B(x))$

Differenz $\forall x \in U : (A \setminus B)(x) = \max(0, A(x) - B(x))$

(mehr dazu und weitere Operationen in den
LV zu Datenbanken)

Für den Spezialfall Mengen (Funktionen nach $\{0, 1\}$)
ergeben $\cup, \cap, \setminus$ genau die Mengenoperationen

Was bisher geschah

Modellierung von

- ▶ **Aussagen** durch Logik

- ▶ **Daten** durch

- ▶ Mengen, Folgen, Sprachen

- ▶ charakteristische Funktionen

- $\chi_A : U \rightarrow \{0, 1\}$ von (Teil-)Mengen $A \subseteq U$

- ▶ Multimengen $A : U \rightarrow \mathbb{N}$ (spezielle Funktionen)

- ▶ Folgen $f : \{0, \dots, n\} \rightarrow B$ (spezielle Funktionen)

je Darstellungsformen, Anwendungen, Beziehungen, Operationen

- ▶ **Zusammenhängen** und **Eigenschaften** durch

- ▶ **Relationen**

- Darstellungsformen, Anwendungen, Beziehungen, Operationen
spezielle binäre Relationen:

- ▶ Äquivalenzrelationen, Zerlegung in Äquivalenzklassen

- ▶ Halbordnungen, Hasse-Diagramm

- ▶ totale Ordnungen, strikte Ordnungen

- ▶ **Funktionen** als spezielle Relationen (WH Mathematik)

- Darstellungsformen, abgeleitete Funktionen, Eigenschaften

- Menge B^A aller (totalen) Funktionen $f : A \rightarrow B$

Anzahl injektiver Funktionen

Wiederholung: Für endliche Mengen A und B ist die Anzahl

- ▶ aller Relationen auf $A \times B$:

$$|\{R \subseteq A \times B\}| = |\{R \in 2^{A \times B}\}| = |2^{A \times B}| = 2^{|A||B|}$$

- ▶ aller totalen Funktionen von A nach B :

$$|\{f : A \rightarrow B\}| = |B^A| = |B|^{|A|}$$

(zum Nachdenken: Spezialfälle $A = \emptyset$ oder $/$ und $B = \emptyset$)

Satz

Die Anzahl aller (totalen und) injektiven Funktionen von einer endlichen Menge A in eine endliche Menge B ist

$$|\{f : A \rightarrow B \mid f \text{ injektiv}\}| = \prod_{i=0}^{|A|-1} (|B| - i)$$

Folgerung

*Für endliche Mengen A und B mit $|A| > |B|$ existiert **keine** (totale und) injektive Funktion $f : A \rightarrow B$.*

Endlicher Schubfachschluss

(pigeonhole principle)

Satz (endlicher Schubfachschluss)

Sind O und S zwei endliche Mengen mit $|O| > |S|$, dann existiert für jede Funktion $f : O \rightarrow S$ ein $s \in S$ mit $|f^{-1}(s)| > 1$.

umgangssprachliche Formulierung:

O – Menge von **Objekten**

S – Menge von **Schubfächern**

Für jede Aufteilung von n Objekten auf $k < n$ Schubfächer existiert ein Schubfach, welches mehr als ein Objekt enthält.

Beispiele

- ▶ Von 13 Personen haben wenigstens zwei im gleichen Monat Geburtstag.
O: 13 Personen, S: 12 Monate
- ▶ Bei 5 Glühwein auf einem Weihnachtsmarkt mit 3 Glühweinständen kommen (wenigstens) zwei Glühwein vom selben Stand.
O: 5 Glühwein, S: 3 Stände
- ▶ 10 verschiedenfarbige Paare Socken
Wieviele blind nehmen für einfarbiges Paar?
O: $x < 20$ Socken, S: 10 Farben
gesucht: kleinstes x mit $x > |S|$ ($x = 11$)
- ▶ je 5 Paare schwarze und weiße Socken
Wieviele blind nehmen für einfarbiges Paar?
O: $x < 20$ Socken, S: 2 Farben ($x = 3$)
- ▶ Partygespräche: $n \geq 2$ Personen
Anzahl der Gesprächspartner ist für zwei Personen gleich.
O: n Gäste, S: Anzahl der Gesprächspartner $\in \{0, \dots, n-1\}$
Problem: $|O| = |S|$
Fallunterscheidung: $\{p_1, \dots, p_n\}$ enthält 0 oder nicht

Erweiterter Schubfachschluss

Satz

Sind O und S zwei endliche Mengen, dann existiert für jede Funktion $f : O \rightarrow S$ ein $s \in S$ mit

$$|f^{-1}(s)| \geq \left\lceil \frac{|O|}{|S|} \right\rceil$$

mit $\lceil x \rceil = \min\{n \in \mathbb{Z} \mid n \geq x\}$ (analog $\lfloor x \rfloor = \max\{n \in \mathbb{Z} \mid n \leq x\}$)

umgangssprachliche Formulierung:

Für jede Aufteilung von n Objekten auf k Schubfächer existiert ein Schubfach, welches mindestens $\lceil \frac{n}{k} \rceil$ Objekte enthält.

Beispiele

- ▶ Unter 30 Personen sind mindestens 5 am gleichen Wochentag geboren.
O: 30 Personen, S: 7 Wochentage
also existiert Wochentag $s \in S$ mit $f^{-1}(s) \geq \lceil \frac{30}{7} \rceil = 5$
- ▶ Von 5 Aussagen haben wenigstens 3 denselben Wahrheitswert.
O: 5 Aussagen, S: $\{0, 1\}$
also existiert Wahrheitswert $s \in \{0, 1\}$ mit $f^{-1}(s) \geq \lceil \frac{5}{2} \rceil = 3$
- ▶ Wieviele Karten höchstens nötig für komplette Farbe aus Skatblatt?
O: $x \leq 32$ Karten, S: 4 Farben, für jede $f^{-1}(s) = 8$
gesucht: kleinstes x mit $7 < \lceil \frac{x}{4} \rceil$, also $x = 29$
- ▶ Unter je 6 Personen existieren immer entweder 3, die einander kennen oder 3, die einander nicht kennen.

Unendlicher Schubfachschluss

endlicher (erweiterter) Schubfachschluss:

Sind O und S zwei endliche Mengen, dann existiert für jede Funktion $f : O \rightarrow S$ ein $s \in S$ mit

$$|f^{-1}(s)| \geq \frac{|O|}{|S|}$$

mit $|O|$ wächst (bei festem S) auch $\frac{|O|}{|S|}$

Satz (unendlicher Schubfachschluß)

Sind O eine unendliche und S eine endliche Menge, dann existiert für jede Funktion $f : O \rightarrow S$ ein $s \in S$ mit unendlichem Urbild $f^{-1}(s)$.

umgangssprachliche Formulierung:

Für jede Aufteilung unendlich vieler Objekte auf endliche viele Schubfächer existiert ein Schubfach, welches unendlich viele Objekte enthält.

Beispiele

- ▶ In jeder unendlichen Menge $A \subseteq \mathbb{N}$ enden unendlich viele Zahlen auf dieselbe Ziffer.
O: A , S: 10 mögliche letzte Stellen
- ▶ In jeder unendlichen Folge von Ziffern kommt (wenigstens) eine Ziffer unendlich oft vor.
O: Folgen-Indices $\mathbb{N}$, S: 10 mögliche Ziffern
- ▶ In jeder unendlichen Folge von Symbolen aus einem endlichen Alphabet A kommt (wenigstens) ein Symbol aus A unendlich oft vor.
O: Folgen-Indices $\mathbb{N}$, S: $|A|$ Symbole
- ▶ In jeder Zahl $r \in \mathbb{R}$ kommt eine Ziffer an unendlich vielen Nachkommastellen vor
O: Nachkommastellen, S: Ziffern $\{0, \dots, 9\}$

Wiederholung: Darstellung binärer Relationen

Darstellungsformen für binäre Relation $R \subseteq M^2$ auf
endlicher Menge M :

- ▶ Menge geordneter Paare (intensional oder extensional)
- ▶ $n \times n$ -Matrix (für $M = \{a_1, \dots, a_n\}$) mit Einträgen aus $\{0, 1\}$

$$\forall i \in \{1, \dots, n\} \forall j \in \{1, \dots, n\} : m_{ij} = \begin{cases} 1 & \text{falls } (a_i, a_j) \in R \\ 0 & \text{sonst} \end{cases}$$

(Charakteristische Funktion der Relation)

- ▶ Diagramm (Graph) mit
 - ▶ Menge M von Ecken und
 - ▶ Menge R von Kanten

wichtige Eigenschaften binärer Relationen:

(ir)reflexiv, transitiv, (a)symmetrisch, antisymmetrisch

Gerichtete Graphen

gerichteter Graph $G = (V, E)$ (Digraph)

- ▶ Menge V von Ecken oder Knoten (vertex)
- ▶ Menge $E \subseteq V^2$ von Kanten (edge)
häufige Notation: ab statt (a, b)

Start- und Endpunkt der Kante $(a, b) \in E$ sind $a, b \in V$

Schlinge Kante $(a, a) \in E$ mit $a \in V$

Beispiele:

- ▶ $G = (V, E)$ mit

$$V = \{0, 1, \dots, 4\}$$

$$E = \{(i, j) \in V^2 \mid j = i + 1 \vee j = i + 3\}$$

- ▶ Abläufe (z.B. Münzspiel)

gerichteter Graph $G = (V, E)$ besteht aus

(endlicher) Menge V mit einer zweistelligen Relation $E \subseteq V^2$

Repräsentationen endlicher Graphen

Graph $G = (V, E)$

Relation $E \subseteq V^2$ (also $E \in 2^{(V^2)}$)
(extensionale oder intensionale) Angabe aller
Elemente in V und E

Diagramm des Graphen

Adjazenzmatrix von $G = (V, E)$ mit $V = \{v_1, \dots, v_n\}$:
 $|V|^2$ -Matrix $M_G \in \{0, 1\}^{|V|^2}$ mit

$$\forall (i, j) \in |V|^2 : M_G(i, j) = \begin{cases} 1 & \text{wenn } (v_i, v_j) \in E \\ 0 & \text{sonst} \end{cases}$$

(charakteristische Funktion der Relation $E \subseteq V^2$)

Adjazenzliste von $G = (V, E)$:
 $L_G : V \rightarrow 2^V$ mit $\forall v \in V : L_G(v) = \pi_2(E|_{\{v\}})$
(d.h. $L_G = \{v \mapsto \{u \in V \mid (v, u) \in E\}\}$)

Ungerichtete Graphen

ungerichteter schlingenfreier Graph (V, E) :

- ▶ Menge V von **Ecken**
- ▶ Menge $E \subseteq \binom{V}{2}$ von **Kanten**
häufige Notation: ab statt $\{a, b\}$

$$\binom{V}{k} = \{M \subseteq V \mid |M| = k\} = \{M \subseteq V \mid M \text{ enthält genau } k \text{ Elemente}\}$$

ungerichteter Graph mit Schlingen: (V, E) mit

$$E \subseteq \binom{V}{2} \cup \binom{V}{1} = \{M \subseteq V \mid |M| = 2 \vee |M| = 1\}$$

Beispiele:

- ▶ $G = (V, E)$ mit $V = \{0, 1, \dots, 4\}$ und
 $E = \left\{ \{i, j\} \in \binom{V}{2} \mid i \neq j \wedge 2 \mid (i - j) \right\}$
- ▶ Liniennetz Nahverkehr

Graph (ohne Zusatz) bedeutet im Folgenden immer

endlich, ungerichtet und schlingenfrei

(Darstellung einer symmetrischen irreflexiven Relation)

Spezielle Graphen

(ungerichtet, schlingenfrei)

Ordnung des Graphen (V, E) : Anzahl $|V|$ der **Ecken**

leerer Graph (V, E) mit $V = \emptyset$ und $E = \emptyset$

Graph (V, E) mit $V = \{v_1, \dots, v_n\}$ heißt

isoliert: $I_n = (\{v_1, \dots, v_n\}, \emptyset)$ (n isolierte Ecken)

vollständig: $K_n = (\{v_1, \dots, v_n\}, E)$
mit $E = \binom{V}{2}$ (n paarweise verbundene Ecken)

Pfad: $P_n = (\{v_1, \dots, v_n\}, E)$
mit $E = \{\{v_i, v_{i+1}\} \mid i \in \{1, \dots, n-1\}\}$

Kreis: $C_n = (\{v_1, \dots, v_n\}, E)$
mit $E = \{\{v_i, v_{i+1}\} \mid i \in \{1, \dots, n-1\}\} \cup \{\{v_n, v_1\}\}$

Was bisher geschah

Modellierung von

- ▶ Aussagen: Aussagenlogik
- ▶ Daten: Mengen, Folgen, Multimengen
- ▶ Beziehungen und Eigenschaften: Relationen, Funktionen

Graphen:

- ▶ $\binom{M}{k}$ für Mengen M (analog Binomialkoeffizienten $\binom{n}{k}$)
- ▶ gerichtete / ungerichtete Graphen $G = (V, E)$
- ▶ Darstellungen von Graphen
- ▶ Spezielle (Isomorphieklassen von) Graphen: I_n, K_n, P_n, C_n

Nachbarschaft in Graphen

Ecken $u, v \in V$ heißen im Graphen (V, E)

benachbart (adjazent) gdw. $uv \in E$

unabhängig gdw. $uv \notin E$

Nachbarschaft (Menge aller Nachbarn) einer Ecke v in G :

$$N_G(v) = \{u \in V \mid uv \in E\}$$

Ecke $v \in V$ mit $N_G(v) = \emptyset$ heißt **isoliert**.

Graph $G = (V, E)$ definiert Funktion $\text{grad}_G : V \rightarrow \mathbb{N}$ mit

$$\forall a \in V : \text{grad}_G(a) = |N_G(a)|$$

$\text{grad}_G(a)$ heißt **Grad** der Ecke a .

(V, E) heißt **n -regulär** (regulär) gdw. $\forall a \in V : \text{grad}_G(a) = n$

Beispiele: alle C_n mit $n > 2$ sind 2-regulär, K_5 ist 4-regulär

Eckengrad in gerichteten Graphen

Nachbarn in gerichteten Graphen $G = (V, E)$:

- ▶ $\forall v \in V : N_{i,G}(v) = \{u \in V \mid uv \in E\}$
(in v **eingehende** Kanten)
- ▶ $\forall v \in V : N_{o,G}(v) = \{u \in V \mid vu \in E\}$
(von v **ausgehende** Kanten)

gerichteter Graph $G = (V, E)$ definiert Funktionen

- ▶ $\text{grad}_{i,G} : V \rightarrow \mathbb{N}$ mit

$$\forall a \in V : \text{grad}_{i,G}(a) = |N_{i,G}(a)|$$

$\text{grad}_{i,G}(a)$ heißt **Eingangs-Grad** der Ecke a .

- ▶ $\text{grad}_{o,G} : V \rightarrow \mathbb{N}$ mit

$$\forall a \in V : \text{grad}_{o,G}(a) = |N_{o,G}(a)|$$

$\text{grad}_{o,G}(a)$ heißt **Ausgangs-Grad** der Ecke a .

Graph-Isomorphie

Ein **Isomorphismus** zwischen den Graphen $G = (V_G, E_G)$ und $H = (V_H, E_H)$ ist eine Bijektion $f : V_G \rightarrow V_H$ mit

$$E_H = \left\{ \{f(x), f(y)\} \in \binom{V_H}{2} \mid \{x, y\} \in E_G \right\}$$

(also $\forall \{x, y\} \in \binom{V_G}{2} : (\{f(x), f(y)\} \in E_H \leftrightarrow \{x, y\} \in E_G)$)

Graphen G und H , zwischen denen ein Isomorphismus existiert, heißen **isomorph** ($G \cong H$).

Beispiel: $G = (\{a, b, c\}, \{\{a, c\}, \{c, b\}\})$ und $H = (\{1, 2, 3\}, \{\{1, 2\}, \{2, 3\}\})$ sind isomorph, denn die Funktion $f : \{a, b, c\} \rightarrow \{1, 2, 3\}$ mit $f = \{a \mapsto 1, b \mapsto 3, c \mapsto 2\}$ ist ein Isomorphismus von G auf H , weil gilt $\{\{f(x), f(y)\} \mid \{x, y\} \in E_G\} = \{\{f(a), f(c)\}, \{f(c), f(b)\}\} = \{\{1, 2\}, \{2, 3\}\} = E_H$

Isomorphie ist eine Äquivalenzrelation auf Menge aller Graphen. (ÜA)

$K_n, I_n, \dots$ bezeichnen Äquivalenzklassen von Graphen bzgl. $\cong$

analog definiert: Definition Isomorphie gerichteter Graphen

Teilgraph-Relationen

Für Graphen $G = (V_G, E_G)$ und $H = (V_H, E_H)$ heißt H

Teilgraph von G , gdw. $V_H \subseteq V_G$ und $E_H \subseteq E_G$

Beispiel: $H = (\{b, c\}, \emptyset)$ ist TG von

$G = (\{a, b, c\}, \{\{a, c\}, \{c, b\}\})$

echter Teilgraph von G , gdw. H Teilgraph von G und $H \neq G$

Beispiel: $H = (\{b, c\}, \emptyset)$ ist echter TG von

$G = (\{a, b, c\}, \{\{a, c\}, \{c, b\}\})$

induzierter Teilgraph von G , gdw. $V_H \subseteq V_G$ und

$E_H = E_G \cap \binom{V_H}{2} = \{\{a, b\} \in E_G \mid \{a, b\} \subseteq V_H\}$

(Autotool: „Beschränkung“ von G auf V_H)

Beispiel: $H = (\{b, c\}, \{\{c, b\}\})$ ist induzierter TG

von $G = (\{a, b, c\}, \{\{a, c\}, \{c, b\}\})$

aufspannender Teilgraph von G , gdw.

H Teilgraph von G und $V_H = V_G$

Beispiel: $H = (\{a, b, c\}, \{\{a, c\}\})$ ist aufspannender

TG von $G = (\{a, b, c\}, \{\{a, c\}, \{c, b\}\})$

Bis auf echter Teilgraph sind diese Relationen Halbordnungen.

Operationen auf Graphen

Operationen für Relationen

auf Graphen $G = (V_G, E_G)$ und $H = (V_H, E_H)$:

$$G \cup H = (V_G \cup V_H, E_G \cup E_H)$$

$$G \cap H = (V_G \cap V_H, E_G \cap E_H)$$

$$G \dot{\cup} H = (V_G \dot{\cup} V_H, E_G \dot{\cup} E_H)$$

$G \cdot_k H$ komponentenweises Produkt

$G \cdot_l H$ lexikographisches Produkt

zusätzliche Operation

$$G * H = (V_G \cup V_H, E_G \cup E_H \cup \{\{u, v\} \mid u \in V_G \wedge v \in V_H\})$$

für V_G und V_H **disjunkt** (join)

(ggf. Knoten umbenennen, d.h. H durch isomorphen Graphen H' mit anderen Knoten ersetzen)

(mehr) spezielle (Isomorphieklassen von) Graphen:

$$K_{m,n} = I_m * I_n \text{ (vollständige bipartite Graphen)}$$

$$K_{1,n} = I_1 * I_n \text{ (Sterne)}$$

$$K_{n_1, \dots, n_m} = I_{n_1} * \dots * I_{n_m} \text{ für } m > 1$$

Komplementärgraph

für $G = (V, E)$ und Eckenmenge $U \subseteq V$

$$G - U = (V \setminus U, E \setminus \{\{u, v\} \mid \{u, v\} \cap U \neq \emptyset\})$$

Für $G = (V, E)$ und Kantenmenge $F \subseteq \binom{V}{2}$

$$G - F = (V, E \setminus F)$$

Komplementärgraph zu $G = (V, E)$:

$$\overline{G} = \left(V, \binom{V}{2} \setminus E \right) = K_{|V|} - E$$

Beispiele: $\overline{I_3} = K_3$, $\overline{C_4} = P_2 \dot{\cup} P_2$, $\overline{P_4} = P_4$

G heißt **selbstkomplementär** gdw. $G \simeq \overline{G}$

Beispiele: I_1, P_4, C_5

Pfade in Graphen

Länge eines Pfades = Anzahl der **Kanten** im Pfad

z.B. Länge von P_5 ist 4

Pfad der Länge n im Graphen G :

Teilgraph von G in der Isomorphieklasse P_{n+1}

Pfad $P = (V', E')$ in $G = (V, E)$ ist eindeutig bestimmt durch

- ▶ Folge $[v_1, \dots, v_n]$ der Ecken in $V' \subseteq V$ oder
- ▶ für Pfade mit wenigstens einer Kante:
Menge $E' \subseteq E$ aller Kanten in P

Maximaler Pfad in G bzgl. Teilgraph-Relation:

Pfad $P = (V', E')$ ist **maximaler** Pfad in $G = (V, E)$ gdw.

für alle Kanten $ab \in E$ gilt:

$(V' \cup \{a, b\}, E' \cup \{ab\})$ ist kein Pfad in G

Pfad von a nach b im Graphen G :

Pfad $p = [v_1, \dots, v_n]$ in G mit $v_1 = a$ und $v_n = b$

Abstand von a und b im Graphen $G = (V, E)$:

$\text{dist}_G(a, b) = \min\{|p| \mid p \in V^* \text{ ist Pfad von } a \text{ und } b \text{ in } G\}$ ($\min \emptyset = \infty$)

(Länge des **kürzesten Pfades** von a nach b in G)

Durchmesser des Graphen $G = (V, E)$:

$\text{diam}(G) = \max\{\text{dist}(a, b) \mid a, b \in V\}$

Kreise und Cliques in Graphen

Länge eines Kreises = Anzahl der **Kanten** im Kreis

z.B. Länge von C_5 ist 5

Kreis der Länge n im Graphen G :

Teilgraph von G in der Isomorphieklasse C_n

echte Kreise: C_n mit $n \geq 3$

Kreis $C = (V', E')$ in $G = (V, E)$ ist eindeutig bestimmt durch

- ▶ Folge $[v_1, \dots, v_n]$ der Ecken in $V' \subseteq V$ oder
- ▶ für Kreise mit wenigstens einer Kante:
Menge $E' \subseteq E$ der Kanten in C

n -Clique im Graphen G :

Teilgraph von G in der Isomorphieklasse K_n

Clique im Graphen G : n -Clique in G für ein $n \in \mathbb{N}$

Cliquenzahl des Graphen G :

$\omega(G) = \max\{n \mid G \text{ enthält eine } n\text{-Clique}\}$

Was bisher geschah

- ▶ gerichtete / ungerichtete Graphen $G = (V, E)$
- ▶ Multigraphen
- ▶ Darstellungen von Graphen
- ▶ Spezielle Graphen: $I_n, K_n, P_n, C_n, K_{m,n}, K_{1,n}, K_{n_1, \dots, n_m}$
- ▶ Beziehungen zwischen Graphen:
Isomorphie,
Teilgraph, induzierter Teilgraph, aufspannender Teilgraph
- ▶ Operationen auf Graphen: $\cup, \cap, *, \overline{}$
- ▶ Nachbarschaften und Eckengrade in Graphen,
 n -reguläre Graphen
- ▶ Pfade, Kreise, Cliques in Graphen

Wege in Graphen

Weg w im Graphen $G = (V, E)$:

Folge von Kanten $w = [v_1 v_2, \dots, v_n v_{n+1}] \in E^*$

(Kanten können mehrfach vorkommen)

durch Folge der Ecken $[v_1, \dots, v_{n+1}]$ eindeutig bestimmt

Weg von a nach b in G :

Weg $[v_1, \dots, v_n]$ in G mit $v_1 = a$ und $v_n = b$

Für jede Ecke $v \in V$ ist $[v]$ ein Weg von v nach v

Wege $[v_1, \dots, v_n]$ mit $v_1 = v_n$ heißen **geschlossen**

Zusammenhang in ungerichteten Graphen

Reflexiv-transitive Hülle $E^* \subseteq V^2$ der Kanten-Relation $E \subseteq V^2$ heißt Zusammenhangsrelation im Graphen $G = (V, E)$

also: $(u, v) \in E^*$ (u und v sind in G zusammenhängend)
gdw. ein Weg von u nach v in G existiert

Für jeden ungerichteten Graphen $G = (V, E)$ ist E^* eine Äquivalenzrelation auf V .

(Äquivalenzklassen $[u]_{E^*}$ sind Mengen von Knoten)

Von den Äquivalenzklassen $[u]_{E^*}$ induzierte Teilgraphen von G heißen **Zusammenhangskomponenten** von G

Graph $G = (V, E)$ heißt **zusammenhängend** gdw.
 G aus genau einer Zusammenhangskomponente besteht.

Beispiele: $P_n, C_n, K_n, K_{m,n}$ sind zusammenhängend,
 I_n für $n > 1$, $\overline{C_4}$ und $P_2 \cup K_4$ nicht

Multigraphen

Wiederholung: ungerichteter Graph $G = (V, E)$ mit $E \subseteq \binom{V}{2}$

z.B. $G = (\{a, b, c, d\}, E)$ mit der charakteristischen Funktion

$$\chi_E : \binom{V}{2} \rightarrow \{0, 1\} \text{ mit } \chi_E(xy) = \begin{cases} 1 & \text{falls } xy \in \{ab, ac, bd, dc\} \\ 0 & \text{sonst} \end{cases}$$

Multigraph $G = (V, E)$ mit **Multimenge** $E : \binom{V}{2} \rightarrow \mathbb{N}$

z.B. $G' = (\{a, b, c, d\}, E')$ mit

$$E'(xy) = \begin{cases} 1 & \text{falls } xy = ab \\ 2 & \text{falls } xy \in \{bd, dc\} \\ 4 & \text{falls } xy = ac \\ 0 & \text{sonst} \end{cases}$$

(analog für gerichtete Graphen mit $E : V^2 \rightarrow \mathbb{N}$ statt $E : V^2 \rightarrow \{0, 1\}$)

Modellierungsbeispiele:

- ▶ Königsberger-Brücken-Problem (Leonhard Euler, 1736)
- ▶ Prozessketten (mit Alternativen zur Bearbeitung von Teilaufgaben)

Eulerwege und -kreise

In $G = (V, E)$ heißt jeder Weg $w = [e_1, \dots, e_n] \in E^*$ genau dann **Eulerweg** in G , wenn er **jede Kante** in E **genau einmal** enthält, d.h.

- ▶ $E = \{e_1, \dots, e_n\}$,
- ▶ $V = \bigcup_{i \in \{1, \dots, n\}} e_i$ und
- ▶ $\forall i \neq j \in \{1, \dots, n\} : e_i \neq e_j$

Eulerkreis (geschlossener Eulerweg) in $G = (V, E)$:
Eulerweg $w = [e_1, \dots, e_n, e_1]$ in G

Satz

1. Ein Graph $G = (V, E)$ enthält genau dann einen geschlossenen Eulerweg, wenn er zusammenhängend ist und $\text{grad}(v)$ für jeden Knoten $v \in V$ gerade ist.
2. Ein zusammenhängender Graph $G = (V, E)$ enthält genau dann einen Eulerweg, wenn er genau zwei oder keine Ecke ungeraden Grades enthält.

Beispiele: Haus vom Nikolaus (+ Varianten), Eulerkreise in $C_n, K_n, K_{m,n}$

Hamiltonpfade und -kreise

Hamiltonpfad in $G = (V, E)$:

Pfad $w = (v_1, \dots, v_n)$ in G mit $V = \{v_1, \dots, v_n\}$
(aufspannender Pfad in G)

Hamiltonkreis in $G = (V, E)$:

Kreis $w = (v_1, \dots, v_n)$ in G mit $V = \{v_1, \dots, v_n\}$
(aufspannender Kreis in G)

Beispiele: Hamiltonkreise in $C_n, K_n, K_{m,n}$

Es ist i.A. aufwendig, festzustellen, ob ein Graph einen Hamiltonkreis enthält.

Anwendungen:

- ▶ Problem des Handlungsreisenden
(TSP: Travelling Salesman Problem)
- ▶ Logistik
- ▶ Pfadplanung, Robotik
- ▶ Hardware-Entwurf

Bipartite Graphen

Ein Graph $G = (V, E)$ heißt **bipartit** gdw. eine Zerlegung $\{V_0, V_1\}$ von V (d.h. $V_0 \cap V_1 = \emptyset$ und $V_0 \cup V_1 = V$) mit $\left(\binom{V_0}{2} \cup \binom{V_1}{2}\right) \cap E = \emptyset$ existiert.

(G also keine Kanten innerhalb der Mengen V_0 und V_1 enthält)

Beispiele:

- ▶ P_n für alle $n \in \mathbb{N} \setminus \{0\}$
- ▶ $K_{m,n}$ für alle $m, n \in \mathbb{N} \setminus \{0\}$
- ▶ kein K_n für $n \in \mathbb{N} \setminus \{0, 1, 2\}$
- ▶ C_n für alle $n \in \mathbb{N} \setminus \{0\}$ mit $2|n$
- ▶ alle Teilgraphen der $K_{m,n}$ für alle $m, n \in \mathbb{N} \setminus \{0\}$

Satz

Ein Graph $G = (V, E)$ ist genau dann bipartit, wenn ein $K_{m,n} = (V, E')$ mit $E \subseteq E'$ existiert.

(d.h., G aufspannender Teilgraph eines $K_{m,n}$ ist)

Anwendung bipartiter Graphen

Modellierung von Zuordnungen, z.B.

- ▶ Personen zu Lieblingsgetränken
- ▶ Getränke-Angebot zu Restaurants
- ▶ Aufgaben zu Mitarbeitern mit unterschiedlichen Qualifikationen
- ▶ Flugzeuge zu Piloten mit Ausbildung auf verschiedenen Flugzeugtypen

Gefärbte (markierte) Graphen

Graph $G = (V, E)$

Menge C_V von Eckenmarkierungen,

Menge C_E von Kantenmarkierungen

Eckenfärbung von G : Zuordnung $f : V \rightarrow C_V$

Kantenfärbung von G : Zuordnung $f : E \rightarrow C_E$

zur Modellierung z.B.:

- ▶ Linennetzplan mit $C_E =$ Menge aller Linien zwischen Stationen
- ▶ Straßennetz mit $C_E =$ Entfernung / Fahrzeit (Kosten) (TSP: Hamilton-Kreis mit geringsten Kosten gesucht)
- ▶ Straßennetz mit $C_E =$ Höchstgeschwindigkeit
- ▶ Spielplan mit $C_V =$ Ereignisfeld-Beschreibung
- ▶ Ablauf-Graphen mit $C_E =$ Options-Auswahl (z.B. Spielzug)
- ▶ Formelbaum mit $C_V =$ Junktoren und Aussagevariablen
- ▶ Multigraphen als Graphen mit $C_E = \mathbb{N} \setminus \{0\}$

Eckenfärbung

Graph $G = (V, E)$, Farben $\{1, \dots, n\}$

$f : V \rightarrow \{1, \dots, k\}$ heißt (konfliktfreie) k -Färbung für G gdw.

$$\forall u \in V \forall v \in V ((uv \in E) \rightarrow \neg(f(u) = f(v)))$$

(also für keine Kante $\{u, v\} \in E$ gilt $f(u) = f(v)$)

Jede Eckenfärbung $f : V \rightarrow \{1, \dots, n\}$ definiert
eine Zerlegung der Eckenmenge V

$$\{f^{-1}(i) \mid i \in \{1, \dots, n\}\}$$

$G = (V, E)$ heißt k -färbbar gdw. k -Färbung für G existiert.

Beispiele:

- ▶ C_5 ist 5-, 4- und 3-färbbar, aber nicht 2-färbbar
- ▶ $K_{3,3}$ ist 2-färbbar,
- ▶ $I_2 * I_2 * I_1$ ist 3-färbbar, aber nicht 2-färbbar

Chromatische Zahl

chromatische Zahl des Graphen G :

$$\chi(G) = \min\{k \mid G \text{ ist } k\text{-färbbar}\}$$

Beispiele:

- ▶ $\chi(C_5) = 3$
- ▶ für alle $n \in \mathbb{N}$ gilt $\chi(K_n) = n$
- ▶ für alle $n \in \mathbb{N}$ gilt $\chi(I_n) = 1$
- ▶ für alle $n > 1$ gilt $\chi(P_n) = 2$
- ▶ für alle $n > 2$ gilt

$$\chi(C_n) = \begin{cases} 2 & \text{falls } n \in 2\mathbb{N} \\ 3 & \text{falls } n \in 2\mathbb{N} + 1 \end{cases}$$

- ▶ für $\min(m, n) \geq 1$ gilt $\chi(K_{m,n}) = 2$

Modellierungsbeispiel Sudoku

Aufgabe informal:

- ▶ 9×9 Felder $\{f_{zs} \mid z \in \{1, \dots, 9\} \wedge s \in \{1, \dots, 9\}\}$
- ▶ einzutragen sind die Ziffern 1 bis 9 in alle leeren Felder
- ▶ Startkonfiguration: einige Felder schon mit Ziffern belegt
- ▶ Bedingungen: keine Zahl mehrfach in
 - ▶ einer Zeile $(s \neq s') \rightarrow (f_{zs} \neq f_{zs'})$
 - ▶ einer Spalte $(z \neq z') \rightarrow (f_{zs} \neq f_{z's})$
 - ▶ einem der neun 3×3 -Blöcke
- ▶ Aufgabe: korrektes Eintragen von Ziffern in jedes freie Feld

Idee: Repräsentation korrekter Lösungen als gefärbte Graphen mit

Felder als Ecken des Graphen

Zahlen $\{1, \dots, 9\}$ als Farben für Knoten

Bedingungen (Konflikte) als Kanten des Graphen

Aufgabe: Finden einer konfliktfreien Färbung des Graphen

Modellierungsbeispiel Sudoku

formal (Verallgemeinerung auf $n^2 \times n^2$ -Feld):

Graph $G = (V, E)$ mit

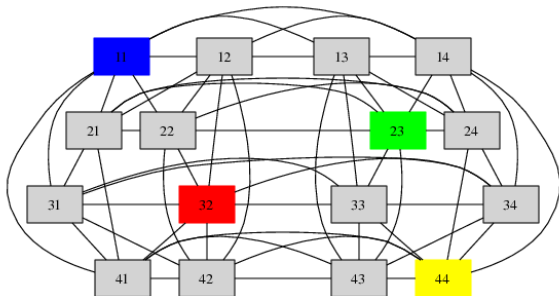
$$V = \{1, \dots, n^2\}^2 \quad (n^2 \times n^2 \text{ Felder})$$

$$C_V = \{1, \dots, n^2\} \quad (n^2 \text{ Zahlen als Farben})$$

$$E = \left\{ \{(z, s), (z', s')\} \in \binom{V}{2} \mid (s = s') \vee (z = z') \vee \left(\left(\lceil \frac{s}{n} \rceil = \lceil \frac{s'}{n} \rceil \right) \wedge \left(\lceil \frac{z}{n} \rceil = \lceil \frac{z'}{n} \rceil \right) \right) \right\}$$

Beispiel für $n = 2$ (also 4×4 -Feld und 4 Farben):

3			
		2	
	4		
			1



Baum-Strukturen

- ▶ Hierarchien
- ▶ Komponenten-Strukturen
- ▶ Familien-Stammbäume
- ▶ Entscheidungsbäume
- ▶ Formeln, arithmetische Ausdrücke

sind spezielle Graphen

- ▶ zusammenhängend
- ▶ enthalten keine Kreise

oft außerdem relevant:

Richtung und Anordnung im Diagramm

Ungerichtete Bäume

Graph $G = (V, E)$ heißt genau dann **Baum**, wenn G

- ▶ zusammenhängend und
- ▶ kreisfrei

(kein Teilgraph von G ist echter Kreis, d.h. C_n für $n \geq 3$)

$v \in V$ mit $\text{grad}(v) \leq 1$ heißt **Blatt**

Fakt

Für jeden Graphen $G = (V, E)$ mit $|E| \geq 2$ und jedes Blatt $v \in V$ gilt: G ist ein Baum gdw. $G - v$ ein Baum ist.

Fakt

Jeder Baum mit mindestens 2 Ecken hat mindestens 2 Blätter.

Fakt

In jedem Baum $G = (V, E)$ gilt $|V| = |E| + 1$

Kreisfreie Graphen $G = (V, E)$ heißen **Wald**.
(Wald = disjunkte Vereinigung von Bäumen)

Charakterisierung der Bäume

Für jeden Graphen $G = (V, E)$ sind folgenden Aussagen äquivalent:

1. G ist ein Baum.
2. Zwischen je zwei Ecken $u, v \in V$ existiert genau ein Pfad in G . (ÜA)
3. G ist minimal zusammenhängend.
(G ist zusammenhängend und für jede Kante $uv \in E$ ist $(V, E \setminus \{uv\})$ nicht zusammenhängend)
4. G ist maximal kreisfrei.
(G enthält keinen echten Kreis und für jede Kante $uv \in \binom{V}{2} \setminus E$ enthält $(V, E \cup \{uv\})$ einen echten Kreis)

Gerüste

Teilgraph H von G heißt **Gerüst** (Spannbaum) von G , falls

- ▶ H ein Baum und
- ▶ H ein aufspannender Teilgraph von G ist.

Beispiele:

- ▶ P_5 in K_5
- ▶ $K_{1,4}$ in K_5
- ▶ G für jeden Baum G
- ▶ I_3 hat kein Gerüst
- ▶ $P_2 \cup K_4$ hat kein Gerüst

Fakt

Ein Graph G ist genau dann zusammenhängend, wenn G ein Gerüst besitzt.

(mehr zur Bestimmung von Gerüsten im Modul
Algorithmen und Datenstrukturen, 2. Semester)

Gerichtete Bäume

Gerichteter Baum:

gerichteter Graph $G = (V, E)$ mit

1. G enthält keinen (ungerichteten) Kreis als Teilgraphen
2. Es existiert genau ein $v \in V$ mit $\text{grad}_{i,G}(v) = 0$ (Wurzel)
3. $\forall u \in V (u \neq v \rightarrow \text{grad}_{i,G}(u) = 1)$

Modellierung durch gerichtete Bäume, z.B.

- ▶ Hierarchien, Klassifikationen
- ▶ Verzeichnis-Strukturen
- ▶ Formelbäume, Termbäume
- ▶ Abstammung
- ▶ Entscheidungsbäume

Häufig ist auch die Ordnung der Kinder relevant.

Gerichtete kreisfreie Graphen (DAG)

Graph $G = (V, E)$ heißt **gerichteter kreisfreier Graph**
(directed acyclic graph, DAG)

gdw. G keinen gerichteten Kreis als Teilgraphen enthält

Quelle $u \in V$ mit $\text{grad}_{i,G} = 0$

Senke $u \in V$ mit $\text{grad}_{o,G} = 0$

Jeder DAG hat wenigstens eine Quelle und eine Senke.

Beispiele:

- ▶ Arithmetische Terme mit gleichen Teiltermen
- ▶ Hasse-Diagramme
- ▶ Abhängigkeiten
- ▶ Binäre Entscheidungsdiagramme (BDDs)
- ▶ Graphen zu Spielverläufen (z.B. Münzenspiel)
- ▶ Abläufe mit unabhängigen Teilaufgaben
- ▶ Programmablaufstrukturen
- ▶ Geschäftsprozesse

Was bisher geschah

Modellierung von

Aussagen durch logische Formeln

Daten durch **Mengen**, Multimengen, Folgen, Sprachen

Zusammenhängen und Eigenschaften von Elementen von Mengen
durch **Relationen**

(Eigenschaften von Relationen)

Spezielle binäre Relationen (QO, ÄR, HO, LO)

Darstellung binärer Relationen durch **Graphen**

Zuordnungen zwischen Elementen von Mengen durch **Funktionen**
(Eigenschaften von Funktionen)

(Mengen,) Multimengen, Folgen als Spezialfälle

Modellierung in Prädikatenlogik

Grundannahme:

Die zu modellierende Welt besteht aus **Individuen**, die **Eigenschaften** haben und zueinander in **Beziehungen** (Relationen, Funktionen) stehen.

Prädikatenlogik zur Formalisierung von Aussagen über Eigenschaften oder Beziehungen von Individuen aus **(algebraischen) Strukturen**

Prädikatenlogische Aussagen

- ▶ Personen sind Geschwister, wenn sie dieselbe Mutter oder denselben Vater haben.
- ▶ A ist genau dann Nachfahre von B, wenn B A's Vater oder A's Mutter ist oder ein Elternteil von A Nachfahre von B ist.
- ▶ Nachfahren derselben Person sind miteinander verwandt.

Individuenbereich: Menge von Personen

Beziehungen: ist-Nachfahre-von, sind-verwandt, sind-Geschwister

Funktionen: Mutter-von, Vater-von

Prädikatenlogische Aussagen

- ▶ Primzahlen sind genau diejenigen natürlichen Zahlen, die genau zwei verschiedene Teiler haben.
- ▶ Gerade Zahlen sind genau diejenigen natürlichen Zahlen, die durch zwei teilbar sind.
- ▶ Es existieren gerade Primzahlen.
- ▶ Nachfolger ungerader Primzahlen sind nicht prim.
- ▶ Das Quadrat jeder geraden Zahl ist gerade.

Individuenbereich: Menge $\mathbb{N}$ aller natürlichen Zahlen

Eigenschaften: prim, gerade

Beziehung: $|$ (teilt)

Funktionen: Nachfolger, Quadrat, 2

(Algebraische) Strukturen – Beispiele

(Träger-)Mengen (Individuenbereiche) mit
Relationen (Eigenschaften, Beziehungen) und
Funktionen (Operationen) auf den Elementen, z.B.

- ▶ Menge aller Personen
Relationen: ist-älter-als, sind-Geschwister (zweistellig)
einstellige Relationen (Eigenschaften): blond
Funktion (einstellig): Mutter-von
- ▶ Menge $\mathbb{N}$ aller natürlichen Zahlen
Relationen: $\geq$ (zweistellig), $|$ (teilt, zweistellig)
einstellige Relationen (Eigenschaften): prim, gerade
Funktionen: Nachfolger (einstellig), $+$ (zweistellig)
- ▶ Menge $\mathbb{R}^2$ aller Punkte der Ebene
Relationen: hat-kleineren-Abstand-von-0 (zweistellig),
bilden-gleichseitiges-Dreieck (dreistellig)
Funktionen: verschieben (einstellig), Mittelpunkt (zweistellig)
- ▶ Menge A^* aller endlichen Wörter über Alphabet A
Relation: Anfangswort, lexikographische Ordnung (zweistellig)
Funktion: Verkettung (zweistellig)

Relationale Strukturen

Eine relationale Struktur $\mathcal{A} = (A, \{R_i \mid i \in I\})$ besteht aus

- ▶ einer nichtleeren Menge A und
- ▶ einer Menge von Relationen $R_i \subseteq A^{n_i}$ (je mit Stelligkeit n_i)

häufige Notation: $(A, R_1, \dots, R_n)$ statt $(A, \{R_1, \dots, R_n\})$

Beispiele:

- ▶ $(\mathbb{N}, |, \leq)$
- ▶ $(\{0, 1\}, \leq, =)$
- ▶ $(2^A, \subseteq)$
- ▶ $(A^*, \sqsubseteq)$
- ▶ jeder Graph (V, E)
- ▶ jeder Graph (V, E, R, G, B) mit
 - ▶ Knotenfärbung (Eigenschaften $R, G \subseteq V$) und
 - ▶ Kantenfärbung (Beziehung $B \subseteq V \times V$)
- ▶ Menge aller endlichen Graphen mit Isomorphie $\cong$ und Teilgraph-Relationen

Strukturen – noch mehr Beispiele

(funktionale) algebraische Strukturen (Algebren) aus den
Mathematik-LV:

Halbgruppe , z.B. $(\mathbb{N}, +)$, $(2\mathbb{Z}, \cdot)$, $(2^X, \cup)$, $(A^*, \circ)$, $(2^{(A^*)}, \circ)$

Monoid , z.B. $(\mathbb{N}, \cdot, 1)$, $(\mathbb{N}, +, 0)$, $(2^X, \cup, \emptyset)$, $(A^*, \circ, \varepsilon)$,
 $(2^{(A^*)}, \circ, \{\varepsilon\})$

Gruppe , z.B. $(\mathbb{Z}, +, 0)$, $(\mathbb{Z}_n, +, 0)$, $(\mathbb{Z}_n, \cdot, 1)$ für Primzahl n

Halbring , z.B. $(2\mathbb{Z}, +, \cdot)$, $(\{0, 1\}, \text{XOR}, \cdot)$, $(2^{(A^*)}, \cup, \circ)$

Ring , z.B. $(\mathbb{Z}, +, \cdot, 0, 1)$, $(\mathbb{Z}_n, +, \cdot, 0, 1)$ für Primzahl n

Körper , z.B. $(\mathbb{Q}, +, \cdot, 0, 1)$, $(\mathbb{Z}_n, +, \cdot, 0, 1)$ für Primzahl n

Vektorraum , z.B. $\mathbb{R}^n$

zwei Trägermengen (Individuen verschiedener Typen):

Skalare $\mathbb{R}$, Vektoren $\mathbb{R}^n$

Funktionen, z.B. skalare Addition, Multiplikation

$+, \cdot : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$

Vektoraddition $+: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^n$, Skalarprodukt

$(\cdot, \cdot) : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$

Kombinationen (Strukturen mit Funktionen und Relationen)

z.B. halbgeordnete Gruppe $(\mathbb{Z}, +, 0, \leq)$

Strukturen desselben Types

$\mathcal{A}$ Menge $\{0, 1\}$ mit

- ▶ Konstanten 0, 1
- ▶ Funktionen min, max (zweistellig)
- ▶ Eigenschaft gerade
- ▶ Relation $\leq$ (zweistellig)

$\mathcal{B}$ Menge aller Studenten im Raum mit

- ▶ Konstanten Anton, Berta
- ▶ Funktionen Älterer (zweistellig),
kleinere-Matrikelnummer (zweistellig)
- ▶ Eigenschaft trägt Brille
- ▶ Relation befreundet (zweistellig)

$\mathcal{C}$ Menge $2^{\mathbb{N}}$ mit

- ▶ Konstanten $\emptyset, \mathbb{N}$
- ▶ Funktionen $\cap, \cup$ (zweistellig)
- ▶ Eigenschaft endlich
- ▶ Relation $\subseteq$ (zweistellig)

Signaturen – Motivation

(syntaktische) Gemeinsamkeiten der Strukturen $\mathcal{A}, \mathcal{B}, \mathcal{C}$:

- ▶ eine Trägermenge
- ▶ zwei Konstanten (nullstellige Funktionen)
- ▶ zwei zweistellige Funktionen
- ▶ eine Eigenschaft (einstellige Relation)
- ▶ eine zweistellige Relation

Notation durch **Symbole** (mit zugeordneter Stelligkeit)
zur Bezeichnung von Relationen und Funktionen

Signaturen

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit Mengen

$\Sigma_F = \{(f, n) \mid n \in \mathbb{N}\}$ von Funktionssymbolen (mit Stelligkeit)

$\Sigma_R = \{(R, n) \mid n \in \mathbb{N}\}$ von Relationssymbolen (mit Stelligkeit)
(nullstellige Funktionssymbole heißen Konstantensymbole)

Signatur definiert einen **Typ** von Strukturen (**Syntax**)

Strukturen mit derselben Signatur unterscheiden sich in der **Semantik** (Bedeutung) der Symbole:

- ▶ Trägermengen
- ▶ Bedeutung der Funktions- und Relationssymbole

Beispiele für Signaturen

- ▶ Signatur für arithmetische Ausdrücke über natürlichen, rationalen, reellen, ... Zahlen
 $\Sigma_F = \{(+, 2), (-, 2), (\cdot, 2), (/ , 2)\} \cup \{(x, 0) \mid x \in \mathbb{N}\}$
(je ein nullstelliges Symbol (Konstante) für jede Zahl aus der Trägermenge, hier z.B. $\mathbb{N}$)
 $\Sigma_R = \emptyset$
- ▶ Signatur für Mengen mit einer zweistelligen Relation (Äquivalenzrelation, Halbordnung, Graph)
 $\Sigma_F = \emptyset, \Sigma_R = \{(R, 2)\}$
- ▶ Signatur für aussagenlogische Formeln in $AL(\{p, q\})$
 $\Sigma_F = \{(\vee, 2), (\wedge, 2), (\neg, 1), (\mathbb{f}, 0), (\mathbb{t}, 0), (p, 0), (q, 0)\},$
 $\Sigma_R = \{(\equiv, 2), (\models, 2), (\text{erfüllbar}, 1)\}$
- ▶ Signatur für alle drei Strukturen $\mathcal{A}, \mathcal{B}, \mathcal{C}$ auf Folie 219
 $\Sigma_F = \{(\clubsuit, 0), (\spadesuit, 0), (\heartsuit, 2), (\diamondsuit, 2)\},$
 $\Sigma_R = \{(\odot, 1), (\mathbb{C}, 2)\}$

Σ -Strukturen

Für eine Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ heißt eine algebraische Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ **Σ -Struktur** gdw.

- ▶ $A \neq \emptyset$ **Trägermenge** (Individuenbereich, Universum)
- ▶ $\forall (f, n) \in \Sigma_F : \llbracket f \rrbracket_{\mathcal{A}} : A^n \rightarrow A$
- ▶ $\forall (R, n) \in \Sigma_R : \llbracket R \rrbracket_{\mathcal{A}} \subseteq A^n$ (bzw. $\llbracket R \rrbracket_{\mathcal{A}} : A^n \rightarrow \{0, 1\}$)

$\llbracket \cdot \rrbracket_{\mathcal{A}}$ ordnet jedem Funktions- und Relationssymbol seine **Bedeutung** in $\mathcal{A}$ (Funktion / Relation passender Stelligkeit) zu.

Beispiel: Σ -Strukturen für die Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit $\Sigma_R = \{(R, 2)\}$ und $\Sigma_F = \{(f, 2), (c, 0)\}$ sind z.B.

- ▶ $\mathcal{A} = (\mathbb{N}, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit
 $\llbracket c \rrbracket_{\mathcal{A}} = 0, \forall (x, y) \in \mathbb{N}^2 : \llbracket f \rrbracket_{\mathcal{A}}(x, y) = x + y,$
 $\llbracket R \rrbracket_{\mathcal{A}} = \{(x, y) \in \mathbb{N}^2 \mid x|y\}$
- ▶ $\mathcal{B} = (\mathbb{Z}, \llbracket \cdot \rrbracket_{\mathcal{B}})$ mit
 $\llbracket c \rrbracket_{\mathcal{B}} = -3, \forall (x, y) \in \mathbb{Z}^2 : \llbracket f \rrbracket_{\mathcal{B}}(x, y) = x \cdot y,$
 $\llbracket R \rrbracket_{\mathcal{B}} = \{(x, y) \in \mathbb{Z}^2 \mid x \leq y\}$

Terme – Syntax

Beispiele: für Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit

$\Sigma_F = \{(3, 0), (5, 0), (+, 2), (\cdot, 2)\}$ und $\Sigma_R = \{(>, 2)\}$ und Menge

$\mathbb{X} = \{x, y, z\}$ von Variablen

- ▶ Ausdrücke $3 + x$, 5 , y , $((y + 5) \cdot (3 \cdot x)) + x$ sind korrekt aus Symbolen aus Σ_F und $\mathbb{X}$ zusammengesetzt
- ▶ Ausdrücke $3x+$, $+5$, $3(x + y)$, $(3x5) + z$ sind nicht korrekt aus Symbolen aus Σ_F und $\mathbb{X}$ zusammengesetzt

Definition (induktiv)

Die Menge $\text{Term}(\Sigma_F, \mathbb{X})$ aller Terme über der (funktionalen) Signatur Σ_F mit Variablen aus der Menge $\mathbb{X}$ ist definiert durch:

IA: $\mathbb{X} \subseteq \text{Term}(\Sigma_F, \mathbb{X})$ (Jede Variable $x \in \mathbb{X}$ ist ein Term.)

IS: Sind $(f, n) \in \Sigma_F$ (f ist n -stelliges Funktionssymbol) und $t_1, \dots, t_n$ Terme aus $\text{Term}(\Sigma_F, \mathbb{X})$,
dann ist $f(t_1, \dots, t_n)$ ein Term aus $\text{Term}(\Sigma_F, \mathbb{X})$.

verschiedene Darstellungen von Termen möglich,

z.B. Baum, Infix-, Präfix-, Postfixform

Grundterme

Terme ohne Variablen heißen **Grundterme**.

$\text{Term}(\Sigma_F, \emptyset)$ ist Menge aller Grundterme über Signatur Σ_F

Beispiele:

für $\Sigma_F = \{(f, 1), (g, 2), (h, 2), (c, 0)\}$ und $\mathbb{X} = \{x, y, z\}$ gilt

- ▶ $c \in \text{Term}(\Sigma_F, \emptyset) \subseteq \text{Term}(\Sigma_F, \mathbb{X})$ (Grundterm)
- ▶ $z \in \text{Term}(\Sigma_F, \mathbb{X})$, aber $z \notin \text{Term}(\Sigma_F, \emptyset)$ (kein Grundterm)
- ▶ $f(c) \in \text{Term}(\Sigma_F, \emptyset) \subseteq \text{Term}(\Sigma_F, \mathbb{X})$
- ▶ $h(f(x), c) \in \text{Term}(\Sigma_F, \mathbb{X})$, aber $h(f(x), c) \notin \text{Term}(\Sigma_F, \emptyset)$
- ▶ $f \notin \text{Term}(\Sigma_F, \mathbb{X})$
- ▶ $h(c) \notin \text{Term}(\Sigma_F, \mathbb{X})$,
- ▶ $x(c) \notin \text{Term}(\Sigma_F, \mathbb{X})$

Warum gilt für alle Signaturen Σ_F ohne Konstantensymbole

$\text{Term}(\Sigma_F, \emptyset) = \emptyset?$ (ÜA)

Beispiele

- ▶ für $\Sigma_F = \{(+, 2), (-, 2), (\cdot, 2), (/ , 2)\} \cup \{(n, 0) \mid n \in \mathbb{N}\}$ ist

$\text{Term}(\Sigma_F, \emptyset)$ die Menge aller arithmetischen Ausdrücke

(Terme) mit Zahlen aus $\mathbb{N}$ (z.B. $3/5+1/4$),

(Achtung: hier geht es nur um die Syntax, Wert i.A. $\notin \mathbb{N}$)

$\text{Term}(\Sigma_F, \{a, b, c\})$ die Menge aller arithmetischen Ausdrücke

(Terme) mit natürlichen Zahlen und Variablen aus der Menge

$\{a, b, c\}$ (z.B. $(3 \cdot a) + ((2 \cdot b)/c)$),

- ▶ $\Sigma_F = \{(\clubsuit, 0), (\spadesuit, 0), (\heartsuit, 2), (\diamondsuit, 2)\}$,
 - ▶ $\clubsuit \in \text{Term}(\Sigma_F, \emptyset)$, Grundterm
 - ▶ $\heartsuit(x, \diamondsuit(y, \spadesuit)) \in \text{Term}(\Sigma_F, \{x, y, z\})$, kein Grundterm
 - ▶ $\spadesuit(\clubsuit(\diamondsuit, \heartsuit(\diamondsuit))) \notin \text{Term}(\Sigma_F, \mathbb{X})$
 - ▶ $\diamondsuit(\clubsuit, \heartsuit(\spadesuit, \clubsuit)) \in \text{Term}(\Sigma_F, \emptyset)$

Größe von Termen

Die Funktion $\text{size}(t) : \text{Term}(\Sigma_F, \mathbb{X}) \rightarrow \mathbb{N}$ ist (induktiv) definiert durch:

IA: falls $t = x \in \mathbb{X}$, dann $\text{size}(t) = 1$

IS: falls $t = f(t_1, \dots, t_n)$, dann

$$\text{size}(t) = 1 + \sum_{i=1}^n \text{size}(t_i)$$

(size ordnet jedem Term $t \in \text{Term}(\Sigma_F, \mathbb{X})$ seine Größe zu)

Beispiele für $\Sigma_F = \{(f, 1), (g, 2), (h, 2), (c, 0)\}$ und $\mathbb{X} = \{x, y, z\}$:

► $\text{size}(x) = 1$ (IA)

► $\text{size}(c) = 1 + 0 = 1$ (IS)

► $\text{size}(f(x)) = 1 + \text{size}(x) = 2$

► $\text{size}(h(f(x), c)) = 1 + \text{size}(f(x)) + \text{size}(c) = 1 + 2 + 1 = 4$

Menge aller Variablen in einem Term

Die Funktion $\text{var}(t) : \text{Term}(\Sigma_F, \mathbb{X}) \rightarrow 2^{\mathbb{X}}$ ist (induktiv) definiert durch:

IA: falls $t = x \in \mathbb{X}$ (Variable), dann $\text{var}(t) = \{x\}$

IS: falls $t = f(t_1, \dots, t_n)$ mit $(f, n) \in \Sigma_F$, dann
$$\text{var}(t) = \text{var}(t_1) \cup \dots \cup \text{var}(t_n)$$

(var ordnet jedem Term $t \in \text{Term}(\Sigma_F, \mathbb{X})$ die Menge aller in t vorkommenden Variablen zu)

Beispiel: Für $\Sigma_F = \{(f, 2), (g, 2), (a, 0)\}$ und
 $t = f(g(x, a), g(f(a, y), x))$
gilt $\text{var}(t) = \{x, y\}$

Wert von Grundtermen in Strukturen

gegeben: (funktionale) Signatur $\Sigma_F = \{(f, n) \mid n \in \mathbb{N}\}$,
 Σ_F -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$

Die Semantik von Grundtermen in Strukturen wird definiert durch die Erweiterung der Semantik-Funktion $\llbracket \cdot \rrbracket_{\mathcal{A}}$ auf Grundterme, d.h. zu einer Funktion $\llbracket \cdot \rrbracket_{\mathcal{A}} : \text{Term}(\Sigma_F, \emptyset) \rightarrow A$

Definition (induktiv)

Der **Wert** des Σ_F -Grundtermes $t = f(t_1, \dots, t_n) \in \text{Term}(\Sigma_F, \emptyset)$ in der Σ_F -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ ist

$$\llbracket t \rrbracket_{\mathcal{A}} = \llbracket f \rrbracket_{\mathcal{A}} (\llbracket t_1 \rrbracket_{\mathcal{A}}, \dots, \llbracket t_n \rrbracket_{\mathcal{A}})$$

Fehlt hier der Induktionsanfang?

nein, ist als Spezialfall enthalten:

Für Terme $t = c$ mit $(c, 0) \in \Sigma_F$ (Konstante) gilt $\llbracket t \rrbracket_{\mathcal{A}} = \llbracket c \rrbracket_{\mathcal{A}}$
(Bedeutung der Konstante c in $\mathcal{A}$, gegeben in Definition von $\mathcal{A}$)

Beispiel

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit

$$\blacktriangleright \Sigma_F = \{(\clubsuit, 0), (\spadesuit, 0), (\heartsuit, 2), (\diamondsuit, 2)\},$$

$$\blacktriangleright \Sigma_R = \{(\odot, 1), (\mathbb{C}, 2)\}$$

$$s = \clubsuit \quad t = \diamondsuit(\clubsuit, \heartsuit(\spadesuit, \clubsuit))$$

Σ -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit

$$A = \mathbb{N}$$

$$\llbracket \clubsuit \rrbracket_{\mathcal{A}} = 5$$

$$\llbracket \spadesuit \rrbracket_{\mathcal{A}} = 3$$

$$\forall (x, y) \in \mathbb{N}^2 : \llbracket \heartsuit \rrbracket_{\mathcal{A}}(x, y) = x + y$$

$$\forall (x, y) \in \mathbb{N}^2 : \llbracket \diamondsuit \rrbracket_{\mathcal{A}}(x, y) = x \cdot y$$

$$\llbracket \odot \rrbracket_{\mathcal{A}} = \{0, \dots, 10\}$$

$$\llbracket \mathbb{C} \rrbracket_{\mathcal{A}} = \{(2n, n) \mid n \in \mathbb{N}\}$$

$$\llbracket s \rrbracket_{\mathcal{A}} = \llbracket \clubsuit \rrbracket_{\mathcal{A}} = 5 \quad \llbracket t \rrbracket_{\mathcal{A}} = \dots$$

Beispiel

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit

- ▶ $\Sigma_F = \{(\clubsuit, 0), (\spadesuit, 0), (\heartsuit, 2), (\diamondsuit, 2)\},$
- ▶ $\Sigma_R = \{(\odot, 1), (\complement, 2)\}$

$$s = \clubsuit \quad t = \diamondsuit(\clubsuit, \heartsuit(\spadesuit, \clubsuit))$$

Σ -Struktur $\mathcal{B} = (B, \llbracket \cdot \rrbracket_{\mathcal{B}})$ mit

$$B = \{0, 1\}$$

$$\llbracket \clubsuit \rrbracket_{\mathcal{B}} = 0$$

$$\llbracket \spadesuit \rrbracket_{\mathcal{B}} = 1$$

$$\forall (x, y) \in \{0, 1\}^2 : \llbracket \heartsuit \rrbracket_{\mathcal{B}}(x, y) = \min(x, y)$$

$$\forall (x, y) \in \{0, 1\}^2 : \llbracket \diamondsuit \rrbracket_{\mathcal{B}}(x, y) = \max(x, y)$$

$$\llbracket \odot \rrbracket_{\mathcal{B}} = \{0\}$$

$$\llbracket \complement \rrbracket_{\mathcal{B}} = \{(0, 0), (0, 1), (1, 1)\}$$

$$\llbracket s \rrbracket_{\mathcal{B}} = \llbracket \clubsuit \rrbracket_{\mathcal{B}} = 0, \quad \llbracket t \rrbracket_{\mathcal{B}} = \dots$$

Weiteres Beispiel

$$s = \clubsuit \quad t = \diamond(\clubsuit, \heartsuit(\spadesuit, \clubsuit))$$

Σ -Struktur $\mathcal{C} = (C, \llbracket \cdot \rrbracket_{\mathcal{C}})$ mit

C = Menge aller Studenten (m/w/d) hier

$\llbracket \clubsuit \rrbracket_{\mathcal{C}}$ = Student rechts vorn

$\llbracket \spadesuit \rrbracket_{\mathcal{C}}$ = Student daneben

$\forall (x, y) \in C^2 : \llbracket \heartsuit \rrbracket_{\mathcal{C}}(x, y) = x$

$\forall (x, y) \in C^2 : \llbracket \diamond \rrbracket_{\mathcal{C}}(x, y) = y$

$\llbracket \odot \rrbracket_{\mathcal{C}} = \{x \in C \mid x \text{ trägt Brille}\}$

$\llbracket \mathbb{C} \rrbracket_{\mathcal{C}} = \{(x, y) \in C^2 \mid x \text{ und } y \text{ im selben SG}\}$

$\llbracket s \rrbracket_{\mathcal{C}} = \llbracket \clubsuit \rrbracket_{\mathcal{C}} = \text{Student rechts vorn} \quad \llbracket t \rrbracket_{\mathcal{C}} = \dots$

Noch ein Beispiel

$$s = \clubsuit \quad t = \diamond(\clubsuit, \heartsuit(\spadesuit, \clubsuit))$$

Σ -Struktur $\mathcal{D} = (D, \llbracket \cdot \rrbracket_{\mathcal{D}})$ mit

$$D = 2^{\mathbb{N}}$$

$$\llbracket \clubsuit \rrbracket_{\mathcal{D}} = \emptyset$$

$$\llbracket \spadesuit \rrbracket_{\mathcal{D}} = \mathbb{N}$$

$$\forall (M, N) \in (2^{\mathbb{N}}) : \llbracket \heartsuit \rrbracket_{\mathcal{D}}(M, N) = M \cap N$$

$$\forall (M, N) \in (2^{\mathbb{N}}) : \llbracket \diamond \rrbracket_{\mathcal{D}}(M, N) = M \cup N$$

$$\llbracket \odot \rrbracket_{\mathcal{D}} = \{M \subseteq \mathbb{N} \mid |M| \in \mathbb{N}\}$$

$$\llbracket \mathbb{C} \rrbracket_{\mathcal{D}} = \{(M, N) \in D^2 \mid M \subseteq N\}$$

$$\llbracket s \rrbracket_{\mathcal{D}} = \llbracket \clubsuit \rrbracket_{\mathcal{D}} = \emptyset, \quad \llbracket t \rrbracket_{\mathcal{D}} = \dots$$

Äquivalenz von Grundtermen in einer Struktur

Σ_F -Grundterme $s, t \in \text{Term}(\Sigma_F, \emptyset)$ mit $\llbracket s \rrbracket_{\mathcal{A}} = \llbracket t \rrbracket_{\mathcal{A}}$ heißen
äquivalent in $\mathcal{A}$ ($s \equiv_{\mathcal{A}} t$).

In den Beispielen oben gilt

$$s \not\equiv_{\mathcal{A}} t$$

$$s \equiv_{\mathcal{B}} t$$

$$s \not\equiv_{\mathcal{C}} t$$

$$s \equiv_{\mathcal{D}} t$$

schon bekannter Spezialfall:
semantische Äquivalenz aussagenlogischer Formeln
(ohne Aussagenvariablen)

Was bisher geschah

(algebraische) **Strukturen** zur zusammenhängenden Modellierung von

Mengen von Individuen (evtl. verschiedener Typen)

Funktionen auf Individuen dieser Mengen

Relationen zwischen Individuen dieser Mengen

klare Unterscheidung zwischen

Syntax: **Symbole** zur formalen (maschinell zu verarbeitenden)
Darstellung von Individuen, Funktionen und Relationen

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$

Terme über der Signatur Σ_F : $\text{Term}(\Sigma_F, \mathbb{X})$
Spezialfall Grundterme (ohne Variablen)

Semantik: Σ -Strukturen definieren **Bedeutung der Symbole** aus Σ
 Σ -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit

- ▶ $A \neq \emptyset$ Trägermenge (Universum)
- ▶ für jedes $(f, n) \in \Sigma_F$ eine Funktion $\llbracket f \rrbracket_{\mathcal{A}} : A^n \rightarrow A$
- ▶ für jedes $(R, n) \in \Sigma_R$ eine Relation $\llbracket R \rrbracket_{\mathcal{A}} \subseteq A^n$

Wiederholung: Signatur, Terme

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit Mengen

$\Sigma_F = \{(f, n) \mid n \in \mathbb{N}\}$ von Funktionssymbolen (mit Stelligkeit)

$\Sigma_R = \{(R, n) \mid n \in \mathbb{N}\}$ von Relationssymbolen (mit Stelligkeit)

Definition (induktiv)

Die Menge $\text{Term}(\Sigma_F, \mathbb{X})$ aller Terme über der (funktionalen) Signatur Σ_F mit Variablen aus der Menge $\mathbb{X}$ ist definiert durch:

IA: Jede Variable $x \in \mathbb{X}$ ist ein Term. $(\mathbb{X} \subseteq \text{Term}(\Sigma_F, \mathbb{X}))$

IS: Sind $(f, n) \in \Sigma_F$ (f ist n -stelliges Funktionssymbol) und $t_1, \dots, t_n$ Terme aus $\text{Term}(\Sigma_F, \mathbb{X})$, dann ist $f(t_1, \dots, t_n)$ ein Term aus $\text{Term}(\Sigma_F, \mathbb{X})$.

WH: Signaturen

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit Mengen

$\Sigma_F = \{(f, n) \mid n \in \mathbb{N}\}$ von Funktionssymbolen (mit Stelligkeit)

$\Sigma_R = \{(R, n) \mid n \in \mathbb{N}\}$ von Relationssymbolen (mit Stelligkeit)
(nullstellige Funktionssymbole heißen Konstantensymbole)

Signatur definiert einen **Typ** von Strukturen (**Syntax**)

Strukturen mit derselben Signatur unterscheiden sich in der **Semantik** (Bedeutung) der Symbole:

- ▶ Trägermengen
- ▶ Bedeutung der Funktions- und Relationssymbole

Σ -Strukturen

Für eine Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ heißt eine algebraische Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ **Σ -Struktur** gdw.

- ▶ $A \neq \emptyset$ **Trägermenge** (Individuenbereich, Universum)
- ▶ $\forall (f, n) \in \Sigma_F : \llbracket f \rrbracket_{\mathcal{A}} : A^n \rightarrow A$
- ▶ $\forall (R, n) \in \Sigma_R : \llbracket R \rrbracket_{\mathcal{A}} \subseteq A^n$ (bzw. $\llbracket R \rrbracket_{\mathcal{A}} : A^n \rightarrow \{0, 1\}$)

$\llbracket \cdot \rrbracket_{\mathcal{A}}$ ordnet jedem Funktions- und Relationssymbol seine **Bedeutung** in $\mathcal{A}$ (Funktion / Relation passender Stelligkeit) zu.

Beispiel: Σ -Strukturen für die Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit $\Sigma_R = \{(R, 2)\}$ und $\Sigma_F = \{(f, 2), (c, 0)\}$ sind z.B.

- ▶ $\mathcal{A} = (\mathbb{N}, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit
 $\llbracket c \rrbracket_{\mathcal{A}} = 0, \forall (x, y) \in \mathbb{N}^2 : \llbracket f \rrbracket_{\mathcal{A}}(x, y) = x + y,$
 $\llbracket R \rrbracket_{\mathcal{A}} = \{(x, y) \in \mathbb{N}^2 \mid x|y\}$
- ▶ $\mathcal{B} = (\mathbb{Z}, \llbracket \cdot \rrbracket_{\mathcal{B}})$ mit
 $\llbracket c \rrbracket_{\mathcal{B}} = -3, \forall (x, y) \in \mathbb{Z}^2 : \llbracket f \rrbracket_{\mathcal{B}}(x, y) = x \cdot y,$
 $\llbracket R \rrbracket_{\mathcal{B}} = \{(x, y) \in \mathbb{Z}^2 \mid x \leq y\}$

WH: Terme

Definition (induktiv)

Die Menge $\text{Term}(\Sigma_F, \mathbb{X})$ aller Terme über der (funktionalen) Signatur Σ_F mit Variablen aus der Menge $\mathbb{X}$ ist definiert durch:

IA: Jede Variable $x \in \mathbb{X}$ ist ein Term. $(\mathbb{X} \subseteq \text{Term}(\Sigma_F, \mathbb{X}))$

IS: Sind $(f, n) \in \Sigma_F$ (f ist n -stelliges Funktionssymbol) und $t_1, \dots, t_n$ Terme aus $\text{Term}(\Sigma_F, \mathbb{X})$,
dann ist $f(t_1, \dots, t_n)$ ein Term aus $\text{Term}(\Sigma_F, \mathbb{X})$.

verschiedene Darstellungen, z.B. Baum, Infix-, Präfix-, Postfixform

Beispiele: für Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit

$\Sigma_R = \{(R, 2)\}$ und $\Sigma_F = \{(f, 2), (c, 0)\}$ $\Sigma_R = \{(R, 2)\}$ und Menge $\mathbb{X} = \{x, y, z\}$ von Variablen

- ▶ Ausdrücke $c, f(c, c), f(f(x, c), y), z, f(f(z, z), f(f(x, c), y))$ sind Terme (korrekt aus Symbolen aus Σ_F und $\mathbb{X}$ zusammengesetzt)
- ▶ Ausdrücke $c(f(y)), y(c), f(x, f(c))$ sind keine Terme (nicht korrekt aus Symbolen aus Σ_F und $\mathbb{X}$ zusammengesetzt)

WH: Wert von Grundtermen in Strukturen

gegeben: funktionale Signatur $\Sigma_F = \{(f, n) \mid n \in \mathbb{N}\}$
 Σ -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$

Definition (induktiv)

Der **Wert** des Σ_F -Grundtermes $t = f(t_1, \dots, t_n) \in \text{Term}(\Sigma_F, \emptyset)$
in der Σ_F -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ ist

$$\llbracket t \rrbracket_{\mathcal{A}} = \llbracket f \rrbracket_{\mathcal{A}} (\llbracket t_1 \rrbracket_{\mathcal{A}}, \dots, \llbracket t_n \rrbracket_{\mathcal{A}})$$

Fehlt hier der Induktionsanfang?

nein, ist als Spezialfall enthalten:

Für Terme $t = c$ mit $(c, 0) \in \Sigma_F$ (Konstante) gilt $\llbracket t \rrbracket_{\mathcal{A}} = \llbracket c \rrbracket_{\mathcal{A}}$
(Bedeutung der Konstante c in $\mathcal{A}$, gegeben in Definition von $\mathcal{A}$)

Interpretation von Termen mit Variablen

gegeben:

- ▶ Signatur $\Sigma = (\Sigma_F, \Sigma_R)$,
- ▶ Variablenmenge $\mathbb{X}$
- ▶ Σ -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$
- ▶ $t \in \text{Term}(\Sigma_F, \mathbb{X})$

Welchen Wert (Individuum) in A haben Variablen $x \in \mathbb{X}$?

Belegung $\beta : \mathbb{X} \rightarrow A$ der Individuenvariablen
(ordnet jeder Variablen einen Wert aus der Trägermenge von $\mathcal{A}$ zu)

Eine **Interpretation** für einen Term $t \in \text{Term}(\Sigma_F, \mathbb{X})$ ist ein Paar
 $(\mathcal{A}, \beta)$ aus

- ▶ einer Σ -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ und
- ▶ einer Belegung $\beta : \mathbb{X} \rightarrow A$.

Beispiele

Term $s = f(g(x, a), g(f(a, y), x))$ über der Signatur $\Sigma = \dots$

- Interpretation $(\mathcal{A}, \alpha)$ mit Σ -Struktur $\mathcal{A} = (\mathbb{N}, \llbracket \cdot \rrbracket_{\mathcal{A}})$, wobei

$$\llbracket a \rrbracket_{\mathcal{A}} = 1$$

$$\forall (m, n) \in \mathbb{N}^2 : \llbracket f \rrbracket_{\mathcal{A}}(m, n) = m + n$$

$$\forall (m, n) \in \mathbb{N}^2 : \llbracket g \rrbracket_{\mathcal{A}}(m, n) = m \cdot n$$

und Variablenbelegung $\alpha : \{x, y\} \rightarrow \mathbb{N}$ mit $\alpha(x) = 2, \alpha(y) = 1$

- Interpretation $(\mathcal{B}, \beta)$ mit Σ -Struktur $\mathcal{B} = (2^{\{a, b, c\}}, \llbracket \cdot \rrbracket_{\mathcal{B}})$:

$$\llbracket a \rrbracket_{\mathcal{B}} = \{b, c\}$$

$$\forall (M, N) \in \left(2^{\{a, b, c\}}\right)^2 : \llbracket f \rrbracket_{\mathcal{B}}(M, N) = M \setminus N$$

$$\forall (M, N) \in \left(2^{\{a, b, c\}}\right)^2 : \llbracket g \rrbracket_{\mathcal{B}}(M, N) = M \cup N$$

und Variablenbelegung $\beta : \{x, y\} \rightarrow 2^{\{a, b, c\}}$ mit
 $\beta(x) = \emptyset, \beta(y) = \{b\}$

Werte von Termen mit Variablen

Der **Wert** des Termes $t \in \text{Term}(\Sigma_F, \mathbb{X})$ in der Σ_F -Interpretation $(\mathcal{A}, \beta)$, bestehend aus

- ▶ der Σ_F -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ und
- ▶ der Belegung $\beta : \mathbb{X} \rightarrow A$

ist (induktiv) definiert durch

IA für $t = x \in \mathbb{X}$ gilt $\llbracket t \rrbracket_{(\mathcal{A}, \beta)} = \beta(x)$

IS für $t = f(t_1, \dots, t_n)$ gilt

$$\llbracket t \rrbracket_{(\mathcal{A}, \beta)} = \llbracket f \rrbracket_{\mathcal{A}} (\llbracket t_1 \rrbracket_{(\mathcal{A}, \beta)}, \dots, \llbracket t_n \rrbracket_{(\mathcal{A}, \beta)})$$

Der Wert eines **Termes** in einer Interpretation $((A, \llbracket \cdot \rrbracket_{\mathcal{A}}), \alpha)$ ist ein **Element aus A** .

(Man bemerke die Analogie zur Berechnung des Wahrheitswertes einer aussagenlogischen Formel mit Aussagenvariablen.)

Der Wert von Grundtermen aus $\text{Term}(\Sigma_F, \emptyset)$ in einer Interpretation $(\mathcal{A}, \alpha)$ hängt nicht von der Belegung α ab.

Beispiele

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit

$\Sigma_F = \{(a, 0), (b, 0), (f, 1), (g, 2), (h, 2)\}$ und $\Sigma_R = \emptyset$

Variablenmenge $\mathbb{X} = \{x, y\}$

Σ -Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}})$ mit

$$S = \mathbb{N}$$

$$\llbracket a \rrbracket_{\mathcal{S}} = 5$$

$$\llbracket b \rrbracket_{\mathcal{S}} = 3$$

$$\forall n \in \mathbb{N} : \llbracket f \rrbracket_{\mathcal{S}}(n) = 1 + n$$

$$\forall (m, n) \in \mathbb{N}^2 : \llbracket g \rrbracket_{\mathcal{S}}(m, n) = m + n$$

$$\forall (m, n) \in \mathbb{N}^2 : \llbracket h \rrbracket_{\mathcal{S}}(m, n) = m \cdot n$$

Variablenbelegungen $\beta : \{x, y\} \rightarrow \mathbb{N}$ mit $\beta(x) = 0, \beta(y) = 1$

$\gamma : \{x, y\} \rightarrow \mathbb{N}$ mit $\gamma(x) = 2, \gamma(y) = 0$

Terme $s = g(h(f(a), x), h(x, y))$ und $t = h(f(x), g(y, a))$

$\llbracket s \rrbracket_{(\mathcal{S}, \beta)} = \dots, \llbracket t \rrbracket_{(\mathcal{S}, \beta)} = \dots, \llbracket s \rrbracket_{(\mathcal{S}, \gamma)} = \dots, \llbracket t \rrbracket_{(\mathcal{S}, \gamma)} = \dots$

Atome in FOL – Syntax

Term repräsentiert **Individuum** aus der Trägermenge

Atom elementare Formel,
repräsentiert **Eigenschaft** von oder **Beziehung**
zwischen Individuen der Trägermenge

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit

- ▶ Menge Σ_F von Funktionssymbolen mit Stelligkeit (f, n)
- ▶ Menge Σ_R von Relationssymbolen mit Stelligkeit (R, n)

Definition

Menge aller **Σ -Atome** mit Variablen aus der Menge $\mathbb{X}$:

$$\text{Atom}(\Sigma, \mathbb{X}) = \{R(t_1, \dots, t_n) \mid (R, n) \in \Sigma_R \text{ und } t_1, \dots, t_n \in \text{Term}(\Sigma_F, \mathbb{X})\}$$

Atome ohne Individuenvariablen heißen **Grundatome**.

Menge aller Grundatome: $\text{Atom}(\Sigma, \emptyset)$

Atome – Beispiele

- ▶ Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit
$$\Sigma_F = \{(1, 0), (2, 0), (\sqrt{}, 1), (+, 2)\}$$
$$\Sigma_R = \{(\leq, 2)\}$$
Variablenmenge $\mathbb{X} = \{x, y, z\}$
 - ▶ $\sqrt{x} = 2 \in \text{Atom}(\Sigma, \mathbb{X})$
 - ▶ $\sqrt{x+2} \notin \text{Atom}(\Sigma, \mathbb{X})$ (aber $\sqrt{x+2} \in \text{Term}(\Sigma_F, \mathbb{X})$)
 - ▶ $\sqrt{1+\sqrt{2}} \leq \sqrt{\sqrt{2}} \in \text{Atom}(\Sigma, \emptyset)$
- ▶ Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit
$$\Sigma_F = \{(a, 0), (b, 0), (k, 2), (d, 2)\}$$
$$\Sigma_R = \{(P, 1), (Q, 2)\}$$
Variablenmenge $\mathbb{X} = \{x, y\}$
 - ▶ $y \notin \text{Atom}(\Sigma, \mathbb{X})$
 - ▶ $P(x) \in \text{Atom}(\Sigma, \mathbb{X})$
 - ▶ $P(a) \in \text{Atom}(\Sigma, \emptyset)$
 - ▶ $P(Q(P(a), k(x, y))) \notin \text{Atom}(\Sigma, \mathbb{X})$
 - ▶ $Q(k(a, d(y, a)), x) \in \text{Atom}(\Sigma, \mathbb{X})$

Atome – Semantik

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$, Variablenmenge $\mathbb{X}$

Σ -Interpretation $(\mathcal{A}, \alpha)$ mit

- ▶ Σ -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$
- ▶ Belegung der Individuenvariablen $\alpha : \mathbb{X} \rightarrow A$

Wert des Atomes $a = P(t_1, \dots, t_n) \in \text{Atom}(\Sigma, \mathbb{X})$
in der Interpretation $(\mathcal{A}, \alpha)$:

$$\begin{aligned}\llbracket a \rrbracket_{(\mathcal{A}, \alpha)} &= \llbracket P(t_1, \dots, t_n) \rrbracket_{(\mathcal{A}, \alpha)} \\ &= \llbracket P \rrbracket_{\mathcal{A}} (\llbracket t_1 \rrbracket_{(\mathcal{A}, \alpha)}, \dots, \llbracket t_n \rrbracket_{(\mathcal{A}, \alpha)}) \\ &= \begin{cases} 1 & \text{falls } (\llbracket t_1 \rrbracket_{(\mathcal{A}, \alpha)}, \dots, \llbracket t_n \rrbracket_{(\mathcal{A}, \alpha)}) \in \llbracket P \rrbracket_{\mathcal{A}} \\ 0 & \text{sonst} \end{cases}\end{aligned}$$

(Spezialfall Atom a mit $(a, 0) \in \Sigma_R$: $\llbracket a \rrbracket_{(\mathcal{A}, \alpha)} = \llbracket a \rrbracket_{\mathcal{A}}$)

Der Wert eines **Atomes** in einer Interpretation $(\mathcal{A}, \alpha)$ ist ein **Wahrheitswert**.

Der Wert von Grundatomen aus $\text{Atom}(\Sigma, \emptyset)$ in einer Interpretation $(\mathcal{A}, \alpha)$ hängt nicht von der Belegung α ab.

Semantik von Atomen – Beispiele

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit

$\Sigma_F = \{(a, 0), (f, 1)\}$ und $\Sigma_R = \{(P, 1), (R, 2)\}$,

Variablenmenge $\mathbb{X} = \{x, y\}$

Σ -Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}})$ mit

$$S = \{0, 1, 2\}$$

$$\llbracket a \rrbracket_{\mathcal{S}} = 1$$

$$\forall d \in S : \llbracket f \rrbracket_{\mathcal{S}}(d) = 2 - d$$

$$\llbracket P \rrbracket_{\mathcal{S}} = \{0, 2\}$$

$$\llbracket R \rrbracket_{\mathcal{S}} = \{(1, 0), (1, 2), (2, 2)\}$$

Belegung $\beta : \{x, y\} \rightarrow \{0, 1, 2\}$: $\beta(x) = 0$, $\beta(y) = 1$

Wert der folgenden Atome aus $\text{Atom}(\Sigma, \mathbb{X})$

in der Interpretation $(\mathcal{S}, \beta)$:

$$\llbracket P(f(x)) \rrbracket_{(\mathcal{S}, \beta)} = \llbracket P \rrbracket_{\mathcal{S}}(\llbracket f(x) \rrbracket_{(\mathcal{S}, \beta)}) = \llbracket P \rrbracket_{\mathcal{S}}(2) = 1 \in \{0, 1\} \text{ (WW)}$$

$$\llbracket P(x) \rrbracket_{(\mathcal{S}, \beta)} = \dots, \llbracket P(a) \rrbracket_{(\mathcal{S}, \beta)} = \dots,$$

$$\llbracket P(f(a)) \rrbracket_{(\mathcal{S}, \beta)} = \dots, \llbracket P(f(f(x))) \rrbracket_{(\mathcal{S}, \beta)} = \dots,$$

$$\llbracket R(x, a) \rrbracket_{(\mathcal{S}, \beta)} = \dots, \llbracket R(f(y), x) \rrbracket_{(\mathcal{S}, \beta)} = \dots$$

Prädikatenlogik (der ersten Stufe) – Syntax (Formeln)

- ▶ aussagenlogische Junktoren $\top, \bot, \neg, \vee, \wedge, \rightarrow, \leftrightarrow$
- ▶ Σ -Atome
- ▶ Quantoren $\forall, \exists$

Die Menge $\text{FOL}(\Sigma, \mathbb{X})$ aller **Formeln der Prädikatenlogik** (der ersten Stufe, first order logic) über der Signatur Σ mit (Individuen-)Variablen aus der Menge $\mathbb{X}$ ist (induktiv) definiert durch:

IA: $\text{Atom}(\Sigma, \mathbb{X}) \subseteq \text{FOL}(\Sigma, \mathbb{X})$

(Alle Atome sind Formeln.)

- IS: ▶ Aus $\{\varphi_1, \dots, \varphi_n\} \subseteq \text{FOL}(\Sigma, \mathbb{X})$ folgt für jeden n -stelligen Junktor $*$:
- $*(\varphi_1, \dots, \varphi_n) \in \text{FOL}(\Sigma, \mathbb{X})$
- ▶ Aus $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$ und $x \in \mathbb{X}$ folgt
- $\{\forall x \varphi, \exists x \varphi\} \subseteq \text{FOL}(\Sigma, \mathbb{X})$

(Baumstruktur, analog zur Definition von Termen)

Beispiele

- Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit
 $\Sigma_F = \{(a, 0), (f, 1)\}$ und $\Sigma_R = \{(P, 1), (R, 2)\}$
Variablenmenge $\mathbb{X} = \{x, y\}$
Formeln aus $\text{FOL}(\Sigma, \mathbb{X})$:

$$\varphi_1 = P(f(x))$$

$$\varphi_2 = (P(x) \leftrightarrow (\neg P(f(x)) \wedge (\forall x \exists y ((P(x) \leftrightarrow P(y)) \wedge R(x, y)))))$$

$$\varphi_3 = \forall x \forall y \varphi_2$$

- $\Sigma = (\emptyset, \Sigma_R)$ mit $\Sigma_R = \{(p, 0), (q, 0)\}$, $\mathbb{X} = \emptyset$
Formeln aus $\text{FOL}(\Sigma, \emptyset)$:

$$\psi_1 = p \vee q$$

$$\psi_2 = \neg p \vee \neg(q \vee p)$$

(Syntax der Aussagenlogik ist Spezialfall der FOL-Syntax)

Modellierung in Prädikatenlogik – Beispiel Geschwister

(Verschiedene) Personen sind (genau dann) Geschwister, wenn sie dieselbe Mutter oder denselben Vater haben.

- ▶ Individuenbereich: Personen
- ▶ Beziehungen: sind-Geschwister (R), Mutter-von (F), Vater-von (F)

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit $\Sigma_F = \{(m, 1), (v, 1)\}$, $\Sigma_R = \{(G, 2), (=, 2)\}$
Zwischenschritte:

- ▶ Die Mutter der Person x ist die Mutter der Person y : $m(x) = m(y)$
- ▶ Die Personen x und y haben dieselbe Mutter oder denselben Vater.
 $m(x) = m(y) \vee v(x) = v(y)$
- ▶ Die Personen x und y sind verschieden und haben dieselbe Mutter oder denselben Vater.
 $\neg(x = y) \wedge (m(x) = m(y) \vee v(x) = v(y))$
- ▶ Die Personen x und y sind genau dann Geschwister, wenn sie verschieden sind und dieselbe Mutter oder denselben Vater haben.
 $G(x, y) \leftrightarrow (\neg(x = y) \wedge (m(x) = m(y) \vee v(x) = v(y)))$
- ▶ (Zwei beliebige) Personen sind genau dann Geschwister, wenn sie verschieden sind und dieselbe Mutter oder denselben Vater haben.

$$\forall x \forall y (G(x, y) \leftrightarrow (\neg(x = y) \wedge (m(x) = m(y) \vee v(x) = v(y))))$$

Variablen in Formeln

Eine Individuenvariable $x \in \mathbb{X}$ **kommt vor** in

Term t , falls $t = x$ oder

$t = f(t_1, \dots, t_n)$ und x in einem t_i vorkommt,

Atom $a = p(t_1, \dots, t_n)$, falls x in einem t_i vorkommt,

Formel φ , falls x in einem Atom in φ

(Teilformel von φ , die ein Atom ist) vorkommt.

var(φ): Menge aller in φ vorkommenden Individuenvariablen

Beispiele:

- ▶ für $\varphi_1 = R(x, a)$ gilt $\text{var}(\varphi_1) = \{x\}$
- ▶ für $\varphi_2 = \exists x R(x, f(y, x))$ gilt $\text{var}(\varphi_2) = \{x, y\}$
- ▶ für $\varphi_3 =$
 $(P(x) \leftrightarrow (\neg P(f(x))) \wedge (\forall x \exists y ((P(x) \leftrightarrow P(y)) \wedge R(x, y))))$
gilt $\text{var}(\varphi_3) = \{x, y\}$

Freie und gebundene Vorkommen von Variablen

Ein **Vorkommen** der Individuenvariablen x in φ heißt

gebunden , falls x in einer Teilformel von φ der Form $\exists x\psi$ oder $\forall x\psi$ vorkommt,

bvar(φ): Menge aller in φ gebunden vorkommenden Variablen

frei , sonst

fvar(φ): Menge aller in φ frei vorkommenden Variablen

$\text{var}(\varphi) = \text{fvar}(\varphi) \cup \text{bvar}(\varphi)$ gilt immer.

$\text{fvar}(\varphi) \cap \text{bvar}(\varphi) \neq \emptyset$ ist möglich, gilt aber nicht immer.

Beispiele

$$\varphi_1 = R(x, a)$$

$$\text{bvar}(\varphi_1) = \emptyset, \text{fvar}(\varphi_1) = \{x\}$$

$$\varphi_2 = \exists x R(x, f(y, x))$$

$$\text{bvar}(\varphi_2) = \{x\}, \text{fvar}(\varphi_2) = \{y\}$$

$$\varphi_3 = (P(x) \leftrightarrow (\neg P(f(x)) \wedge (\forall x \exists y ((P(x) \leftrightarrow P(y)) \wedge R(x, y)))))$$

$$\text{bvar}(\varphi_3) = \{x, y\}, \text{fvar}(\varphi_3) = \{x\}$$

$$\varphi_4 = \forall x \forall y \varphi_3$$

$$\text{bvar}(\varphi_4) = \{x, y\}, \text{fvar}(\varphi_4) = \emptyset$$

Sätze

Jede Formel $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$ mit $\text{fvar}(\varphi) = \emptyset$ heißt **Satz**.

Beispiele: Für $\Sigma = (\Sigma_F, \Sigma_R)$ mit $\Sigma_F = \{(c, 0)\}$, $\Sigma_R = \{(R, 2)\}$ und $\mathbb{X} = \{x, y\}$ sind

- ▶ $\forall x \forall y (R(x, y) \rightarrow R(y, x))$ ein Satz in $\text{FOL}(\Sigma, \mathbb{X})$,
- ▶ $R(c, c)$ ein Satz in $\text{FOL}(\Sigma, \mathbb{X})$,
- ▶ $\exists y (R(c, y) \rightarrow R(y, c))$ ein Satz in $\text{FOL}(\Sigma, \mathbb{X})$,
- ▶ $R(x, y) \rightarrow R(y, x)$,
 $\forall x (R(x, y) \rightarrow R(y, x))$,
 $\exists y (R(x, y) \rightarrow R(y, x))$,
 $\forall x (R(x, y) \rightarrow \exists y R(y, x))$ keine Sätze in $\text{FOL}(\Sigma, \mathbb{X})$.

für $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$ mit $\text{fvar}(\varphi) = \{x_1, \dots, x_n\}$:

universeller Abschluss von φ : $\forall \varphi := \forall x_1 \cdots \forall x_n \varphi$

existentieller Abschluss von φ : $\exists \varphi := \exists x_1 \cdots \exists x_n \varphi$

Für jede Formel $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$ sind $\forall \varphi$ und $\exists \varphi$ Sätze.

Was bisher geschah

Klassische Prädikatenlogik der ersten Stufe $\text{FOL}(\Sigma, \mathbb{X})$:

- Syntax**
- ▶ Signatur $\Sigma = (\Sigma_F, \Sigma_R)$,
 - ▶ Menge $\mathbb{X}$ von (Individuen-)Variablen
 - ▶ Terme $\text{Term}(\Sigma, \mathbb{X})$, Grundterme $\text{Term}(\Sigma, \emptyset)$
 - ▶ Atome $\text{Atom}(\Sigma, \mathbb{X})$, Grundatome $\text{Atom}(\Sigma, \emptyset)$
 - ▶ Formeln $\text{FOL}(\Sigma, \mathbb{X})$,
 - ▶ Sätze,
 - ▶ universeller, existentieller Abschluss von φ : $\forall\varphi, \exists\varphi$

- Semantik**
- ▶ Σ -Strukturen $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$
 - ▶ Belegung der Individuenvariablen $\alpha : \mathbb{X} \rightarrow A$
 - ▶ Σ -Interpretationen $(\mathcal{A}, \alpha)$

Werte von

- ▶ (Individuen-)Variablen aus $\mathbb{X}$,
- ▶ Termen aus $\text{Term}(\Sigma, \mathbb{X})$,
- ▶ Atomen aus $\text{Atom}(\Sigma, \mathbb{X})$

in Σ -Interpretationen

Beispiel

$$\varphi = \neg R(x, y)$$

Interpretation $(\mathcal{S}, \beta)$ mit

Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}})$ mit $S = \{1, 2, 3\}$ und

$$\llbracket R \rrbracket_{\mathcal{S}} = \{(1, 3), (2, 2), (2, 3), (3, 1)\}$$

Werte $\llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)}$ für Belegungen $\beta : \{x, y\} \rightarrow S$
mit $\beta(y) = 3$ und alle möglichen Werte für $\beta(x) \in S$:

- ▶ $\beta(x) = 1$: $\llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)} = 0$ wegen $(1, 3) \in \llbracket R \rrbracket_{\mathcal{S}}$
- ▶ $\beta(x) = 2$: $\llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)} = 0$ wegen $(2, 3) \in \llbracket R \rrbracket_{\mathcal{S}}$
- ▶ $\beta(x) = 3$: $\llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)} = 1$ wegen $(3, 3) \notin \llbracket R \rrbracket_{\mathcal{S}}$

$$\max \{ \llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)} \mid \beta(x) \in S \} = 1$$

$$\min \{ \llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)} \mid \beta(x) \in S \} = 0$$

Modifizierte Interpretationen

gegeben:

- ▶ Signatur Σ ,
- ▶ Variablenmenge $\mathbb{X}$,
- ▶ Σ -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$,
- ▶ Belegung $\beta : \mathbb{X} \rightarrow A$ der Variablen aus $\mathbb{X}$ in A

modifizierte Belegung $\beta[x \mapsto d] : \mathbb{X} \rightarrow A$ mit $d \in A$:

$$\beta[x \mapsto d](y) = \begin{cases} d & \text{für } y = x \\ \beta(y) & \text{sonst} \end{cases}$$

modifizierte Interpretation $(\mathcal{A}, \beta[x \mapsto d])$

Beispiel:

- ▶ Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit $\dots$, Variablenmenge $\mathbb{X} = \{x, y\}$
- ▶ Σ -Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}})$ mit $S = \{0, 1, 2\}, \dots$
- ▶ Belegung $\beta : \{x, y\} \rightarrow \{0, 1, 2\}$ mit $\beta(x) = 0, \beta(y) = 2$
- ▶ modifizierte Belegung $\beta[x \mapsto 1] : \{x, y\} \rightarrow \{0, 1, 2\}$ mit $\beta[x \mapsto 1](x) = 1, \beta[x \mapsto 1](y) = \beta(y) = 2$
- ▶ modifizierte Interpretation $(\mathcal{S}, \beta[x \mapsto 1])$

Wert in Interpretationen – Semantik FOL

Wert in Σ -Interpretation $(\mathcal{S}, \beta)$ mit $\mathcal{S} = (\mathcal{S}, \llbracket \cdot \rrbracket_{\mathcal{S}})$

- ▶ einer **Individuenvariable** $x \in \mathbb{X}$: $\llbracket x \rrbracket_{(\mathcal{S}, \beta)} = \beta(x) \in \mathcal{S}$
- ▶ eines **Termes** $t = f(t_1, \dots, t_n) \in \text{Term}(\Sigma_F, \mathbb{X})$:
 $\llbracket t \rrbracket_{(\mathcal{S}, \beta)} = \llbracket f \rrbracket_{\mathcal{S}} (\llbracket t_1 \rrbracket_{(\mathcal{S}, \beta)}, \dots, \llbracket t_n \rrbracket_{(\mathcal{S}, \beta)}) \in \mathcal{S}$
- ▶ eines **Atomes** $a = p(t_1, \dots, t_n) \in \text{Atom}(\Sigma, \mathbb{X})$:
 $\llbracket a \rrbracket_{(\mathcal{S}, \beta)} = \llbracket p \rrbracket_{\mathcal{S}} (\llbracket t_1 \rrbracket_{(\mathcal{S}, \beta)}, \dots, \llbracket t_n \rrbracket_{(\mathcal{S}, \beta)}) \in \{0, 1\}$
- ▶ einer **Formel** $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$: $\llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)} \in \{0, 1\}$

$$\begin{aligned}\llbracket \neg \varphi \rrbracket_{(\mathcal{S}, \beta)} &= 1 - \llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)} \\ \llbracket \varphi \vee \psi \rrbracket_{(\mathcal{S}, \beta)} &= \max(\llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)}, \llbracket \psi \rrbracket_{(\mathcal{S}, \beta)}) \\ \llbracket \varphi \wedge \psi \rrbracket_{(\mathcal{S}, \beta)} &= \min(\llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)}, \llbracket \psi \rrbracket_{(\mathcal{S}, \beta)}) \\ \llbracket \exists x \varphi \rrbracket_{(\mathcal{S}, \beta)} &= \max\{\llbracket \varphi \rrbracket_{(\mathcal{S}, \beta[x \mapsto a])} \mid a \in \mathcal{S}\} \\ \llbracket \forall x \varphi \rrbracket_{(\mathcal{S}, \beta)} &= \min\{\llbracket \varphi \rrbracket_{(\mathcal{S}, \beta[x \mapsto a])} \mid a \in \mathcal{S}\}\end{aligned}$$

(zur Semantik des Allquantors siehe auch

<https://web.archive.org/web/20201212021712/spikedmath.com/445.html>)

Beispiele

$$\Sigma = (\Sigma_R, \Sigma_F) \text{ mit } \Sigma_F = \{(f, 1)\}, \Sigma_R = \{(R, 2)\}, \mathbb{X} = \{x, y, z\}$$

$$\varphi = \neg R(x, y) \wedge \exists z R(z, z) \quad \psi = \forall x \exists y R(x, f(y))$$

- ▶ Σ -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit $A = \{a, b, c\}$

$$\llbracket f \rrbracket_{\mathcal{A}}(a) = \llbracket f \rrbracket_{\mathcal{A}}(b) = c, \llbracket f \rrbracket_{\mathcal{A}}(c) = a$$

$$\llbracket R \rrbracket_{\mathcal{A}} = \{(a, a), (b, a), (a, c)\}$$

$$\text{Belegung } \alpha(x) = b, \alpha(y) = a, \alpha(z) = b$$

$$\llbracket \varphi \rrbracket_{(\mathcal{A}, \alpha)} = \dots$$

$$\llbracket \psi \rrbracket_{(\mathcal{A}, \alpha)} = \dots$$

- ▶ Σ -Struktur $\mathcal{B} = (B, \llbracket \cdot \rrbracket_{\mathcal{B}})$ mit $B = \mathbb{Z}$

$$\llbracket f \rrbracket_{\mathcal{B}}(d) = -d, \llbracket R \rrbracket_{\mathcal{B}} = \leq$$

$$\text{Belegung } \beta(x) = 5, \beta(y) = 3, \beta(z) = -1,$$

$$\llbracket \varphi \rrbracket_{(\mathcal{B}, \beta)} = \dots$$

$$\llbracket \psi \rrbracket_{(\mathcal{B}, \beta)} = \dots$$

Modelle für Formeln

Σ -Interpretation (S, β) **erfüllt** (ist Modell für) die Formel $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$ genau dann, wenn $\llbracket \varphi \rrbracket_{(S, \beta)} = 1$.

Menge aller Modelle der Formel $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$

$$\text{Mod}(\varphi) = \left\{ (S, \beta) \mid \begin{array}{l} \mathcal{S} = (S, \llbracket \cdot \rrbracket_S) \text{ ist } \Sigma\text{-Struktur und } \beta : \mathbb{X} \rightarrow S \\ \text{und } \llbracket \varphi \rrbracket_{(S, \beta)} = 1 \end{array} \right\}$$

Formel $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$ heißt

erfüllbar gdw. $\text{Mod}(\varphi) \neq \emptyset$ (z.B. $\forall x (P(x) \vee Q(x))$)

unerfüllbar gdw. $\text{Mod}(\varphi) = \emptyset$ (z.B. $\forall x P(x) \wedge \exists x \neg P(x)$)

allgemeingültig gdw.

$$\text{Mod}(\varphi) = \{(S, \beta) \mid \mathcal{S} = (S, \llbracket \cdot \rrbracket_S) \text{ ist } \Sigma\text{-Struktur und } \beta : \mathbb{X} \rightarrow S\}$$

(Menge aller Σ -Interpretationen)

Beispiele

$$\varphi = R(x) \wedge \exists y((\neg R(y)) \wedge E(y, x))$$

- ▶ Interpretation $(\mathcal{G}, \alpha)$ aus Struktur $\mathcal{G} = (\{1, \dots, 4\}, \llbracket \cdot \rrbracket_{\mathcal{G}})$ mit
 $\llbracket R \rrbracket_{\mathcal{G}} = \{1, 2, 4\}$,
 $\llbracket E \rrbracket_{\mathcal{G}} = \{(1, 2), (3, 2)\}$ und
und Belegung $\alpha : \{x, y\} \rightarrow \{1, \dots, 4\}$ mit $\alpha(x) = 2, \alpha(y) = 2$
 $(\mathcal{G}, \alpha) \in \text{Mod}(\varphi)$
- ▶ Interpretation $(\mathcal{H}, \beta)$ aus Struktur $\mathcal{H} = (\{a, b\}, \llbracket \cdot \rrbracket_{\mathcal{H}})$ mit
 $\llbracket R \rrbracket_{\mathcal{H}} = \{a\}$,
 $\llbracket E \rrbracket_{\mathcal{H}} = \{(a, a), (b, b)\}$,
und Belegung $\beta : \{x, y\} \rightarrow \{a, b\}$ mit $\beta(x) = a, \beta(y) = a$
 $(\mathcal{H}, \beta) \notin \text{Mod}(\varphi)$
- ▶ Interpretation $(\mathcal{J}, \gamma)$ aus Struktur $\mathcal{J} = (\mathbb{Z}, \llbracket \cdot \rrbracket_{\mathcal{J}})$ mit
 $\llbracket R \rrbracket_{\mathcal{J}} = 2\mathbb{Z}$ (Menge aller geraden Zahlen),
 $\llbracket E \rrbracket_{\mathcal{J}} = \{(i, i+1) \mid i \in \mathbb{Z}\}$ und
und Belegung $\gamma : \{x, y\} \rightarrow \mathbb{Z}$ mit $\gamma(x) = \gamma(y) = 0$
 $(\mathcal{J}, \gamma) \in \text{Mod}(\varphi)$

Semantische Äquivalenz von Formeln

(analog Aussagenlogik)

Für $\varphi, \psi \in \text{FOL}(\Sigma, \mathbb{X})$ gilt

$$\varphi \equiv \psi \quad \text{gdw.} \quad \text{Mod}(\varphi) = \text{Mod}(\psi)$$

Beispiele:

- ▶ $\neg \forall x P(x) \equiv \exists x \neg P(x)$
(d.h.: für alle Σ -Interpretationen $(\mathcal{S}, \beta)$ gilt
 $\llbracket \neg \forall x P(x) \rrbracket_{(\mathcal{S}, \beta)} = \llbracket \exists x \neg P(x) \rrbracket_{(\mathcal{S}, \beta)}$)
- ▶ Für alle Formeln $\varphi, \psi \in \text{FOL}(\Sigma, \mathbb{X})$ gilt
 $\exists x \varphi \vee \exists x \psi \equiv \exists x (\varphi \vee \psi)$
- ▶ $\forall x \exists y R(x, y) \not\equiv \forall x \exists y R(y, x)$,
(d.h.: es gibt Interpretationen $(\mathcal{S}, \beta)$, mit
 $\llbracket \forall x \exists y R(x, y) \rrbracket_{(\mathcal{S}, \beta)} \neq \llbracket \forall x \exists y R(y, x) \rrbracket_{(\mathcal{S}, \beta)}$)
- ▶ $\forall x \exists y R(x, y) \not\equiv \exists x \forall y R(x, y)$

Beweis für $\neg\forall xP(x) \equiv \exists x\neg P(x)$

zu zeigen:

$$\text{Mod}(\neg\forall xP(x)) = \text{Mod}(\exists x\neg P(x))$$

(Für alle Σ -Interpretationen $(\mathcal{S}, \beta)$ gilt

$$\llbracket \neg\forall xP(x) \rrbracket_{(\mathcal{S}, \beta)} = \llbracket \exists x\neg P(x) \rrbracket_{(\mathcal{S}, \beta)}$$

$$\begin{aligned}\text{Mod}(\neg\forall xP(x)) &= \{(\mathcal{S}, \beta) \mid \llbracket \neg\forall xP(x) \rrbracket_{(\mathcal{S}, \beta)} = 1\} \\&= \{(\mathcal{S}, \beta) \mid \llbracket \forall xP(x) \rrbracket_{(\mathcal{S}, \beta)} = 1 - 1 = 0\} \\&= \{(\mathcal{S}, \beta) \mid \min\{\llbracket P(x) \rrbracket_{(\mathcal{S}, \beta[x \mapsto d])} \mid d \in S\} = 0\} \\&= \{(\mathcal{S}, \beta) \mid \exists d \in S : \llbracket P(x) \rrbracket_{(\mathcal{S}, \beta[x \mapsto d])} = 0\} \\&= \{(\mathcal{S}, \beta) \mid \exists d \in S : 1 - \llbracket P(x) \rrbracket_{(\mathcal{S}, \beta[x \mapsto d])} = 1\} \\&= \{(\mathcal{S}, \beta) \mid \exists d \in S : \llbracket \neg P(x) \rrbracket_{(\mathcal{S}, \beta[x \mapsto d])} = 1\} \\&= \{(\mathcal{S}, \beta) \mid \max\{\llbracket \neg P(x) \rrbracket_{(\mathcal{S}, \beta[x \mapsto d])} \mid d \in S\} = 1\} \\&= \{(\mathcal{S}, \beta) \mid \llbracket \exists x\neg P(x) \rrbracket_{(\mathcal{S}, \beta)} = 1\} \\&= \text{Mod}(\exists x\neg P(x))\end{aligned}$$

Wichtige Äquivalenzen mit Quantoren

Für alle Formeln $\varphi, \psi \in \text{FOL}(\Sigma, \mathbb{X})$ gilt

$$\neg \forall x \varphi \equiv \exists x \neg \varphi$$

$$\neg \exists x \varphi \equiv \forall x \neg \varphi$$

$$\forall x \varphi \wedge \forall x \psi \equiv \forall x (\varphi \wedge \psi)$$

$$\exists x \varphi \vee \exists x \psi \equiv \exists x (\varphi \vee \psi)$$

$$\forall x \forall y \varphi = \forall y \forall x \varphi$$

$$\exists x \exists y \varphi = \exists y \exists x \varphi$$

falls $x \notin \text{fvar}(\psi)$ und $* \in \{\vee, \wedge\}$, gilt außerdem

$$\forall x \varphi * \psi \equiv \forall x (\varphi * \psi)$$

$$\exists x \varphi * \psi \equiv \exists x (\varphi * \psi)$$

$$\exists y \psi \equiv \exists x \psi_{[y \mapsto x]}$$

$$\forall y \psi \equiv \forall x \psi_{[y \mapsto x]}$$

Notation $\psi_{[y \mapsto x]}$ in den letzten Zeilen:

alle in ψ freien Vorkommen von y durch x ersetzt

Beweis für $\exists y\psi \equiv \exists x\psi_{[y \mapsto x]}$

zu zeigen: Falls $x \notin \text{fvar}(\psi)$ gilt

$$\text{Mod}(\exists y\psi) = \text{Mod}(\exists x\psi_{[y \mapsto x]})$$

$$\begin{aligned}\text{Mod}(\exists y\psi) &= \{(\mathcal{S}, \beta) \mid \llbracket \exists y\psi \rrbracket_{(\mathcal{S}, \beta)} = 1\} \\ &= \{(\mathcal{S}, \beta) \mid \max\{\llbracket \psi \rrbracket_{(\mathcal{S}, \beta[y \mapsto d])} \mid d \in S\} = 1\} \\ &= \{(\mathcal{S}, \beta) \mid \max\{\llbracket \psi_{[y \mapsto x]} \rrbracket_{(\mathcal{S}, \beta[x \mapsto d])} \mid d \in S\} = 1\} \\ &= \{(\mathcal{S}, \beta) \mid \llbracket \exists x\psi_{[y \mapsto x]} \rrbracket_{(\mathcal{S}, \beta)} = 1\} \\ &= \text{Mod}(\exists x\psi_{[y \mapsto x]})\end{aligned}$$

Die Bedingung $x \notin \text{fvar}(\psi)$ ist notwendig.

Beispiel:

Für $\psi = R(x, y)$ gilt $\psi_{[y \mapsto x]} = R(x, x)$, aber $\exists yR(x, y) \not\equiv \exists xR(x, x)$

Nachweis durch Gegenbeispiel: In der Interpretation $(\mathcal{S}, \beta)$ mit

Struktur $\mathcal{S} = (\{a, b\}, \llbracket \cdot \rrbracket_{\mathcal{S}})$ und $\llbracket R \rrbracket_{\mathcal{S}} = \{(a, b)\}$

und Belegung $\beta : \{x, y\} \rightarrow \{a, b\}$ mit $\beta(x) = a$ und $\beta(y) = b$

gilt $\llbracket \exists yR(x, y) \rrbracket_{(\mathcal{S}, \beta)} = 1 \neq 0 = \llbracket \exists xR(x, x) \rrbracket_{(\mathcal{S}, \beta)}$

Was bisher geschah

Daten durch Mengen (Individuenbereiche),
häufig strukturiert, zusammengesetzt durch Operationen

Zusammenhängen zwischen Individuen durch

Relationen (auch Eigenschaften)

Funktionen (Operationen)

gemeinsam in (algebraischen) Strukturen

Bedingungen (Eigenschaften, Anforderungen) an Individuen und deren
Beziehungen durch Formel(menge)n in **Logik** $\text{FOL}(\Sigma, \mathbb{X})$

Syntax

- ▶ Signatur, Variablen
- ▶ Terme (induktive Definition)
- ▶ Atome (in $\text{AL}(P)$: Aussagevariablen)
- ▶ Formeln (Junktoren, Quantoren),
Sätze

Semantik

- ▶ Σ -Struktur $\mathcal{S}$, Σ -Interpretation $(\mathcal{S}, \beta)$
- ▶ Wert von Termen, Atomen, Formeln
in Σ -Interpretationen
- ▶ Modellmengen von Formeln,
Formelmengen, Sätzen, Satzungen
- ▶ erfüllbar, unerfüllbar, allgemeingültig

Modellierung in Prädikatenlogik – Beispiel Topfdeckel

Aussage: Auf jeden Topf passt ein Deckel.

- ▶ Individuenbereich: Kochgeschirr
- ▶ Eigenschaften: ist-Topf $T(\cdot)$, ist-Deckel $D(\cdot)$
- ▶ Beziehung: passt-auf $P(\cdot, \cdot)$

Schrittweise Entwicklung einer Formel zur Aussage:

1. Atome: $D(x)$ (x ist ein Deckel), $T(y)$ (y ist ein Topf),
 $P(x, y)$ (x passt auf y)
2. Formel $D(x) \wedge P(x, y)$
 x ist ein Deckel und passt auf (das Individuum) y .
3. Formel $\exists x (D(x) \wedge P(x, y))$
Es gibt einen Deckel, welcher auf y passt.
4. Formel $T(y) \rightarrow \exists x (D(x) \wedge P(x, y))$
Wenn y ein Topf ist, dann gibt einen Deckel, der auf y passt.
5. Formel $\forall y (T(y) \rightarrow \exists x (D(x) \wedge P(x, y)))$
Zu jedem Topf gibt es einen Deckel, der auf diesen Topf passt.
(dieselbe Bedeutung wie die Aussage oben)

WH: Modelle von Formeln – Beispiele

$$\varphi = R(x) \wedge \exists y((\neg R(y)) \wedge E(y, x))$$

- ▶ Struktur $\mathcal{G} = (\{1, \dots, 4\}, \llbracket \cdot \rrbracket_{\mathcal{G}})$ mit
 $\llbracket R \rrbracket_{\mathcal{G}} = \{1, 2, 4\}$,
 $\llbracket E \rrbracket_{\mathcal{G}} = \{(1, 2), (3, 2)\}$ und
 Belegung $\alpha : \{x, y\} \rightarrow \{1, \dots, 4\}$ mit $\alpha(x) = 2, \alpha(y) = 2$
 $(\mathcal{G}, \alpha) \in \text{Mod}(\varphi)$
- ▶ Struktur $\mathcal{H} = (\{a, b\}, \llbracket \cdot \rrbracket_{\mathcal{H}})$ mit
 $\llbracket R \rrbracket_{\mathcal{H}} = \{a\}$,
 $\llbracket E \rrbracket_{\mathcal{H}} = \{(a, a), (b, b)\}$,
 Belegung $\beta : \{x, y\} \rightarrow \{a, b\}$ mit $\beta(x) = a, \beta(y) = a$
 $(\mathcal{H}, \beta) \notin \text{Mod}(\varphi)$

Modelle von Formelmengen

Menge aller Modelle der Formelmenge $\Phi \subseteq \text{FOL}(\Sigma, \mathbb{X})$:

$$\text{Mod}(\Phi) = \bigcap_{\varphi \in \Phi} \text{Mod}(\varphi)$$

Beispiel: Signatur $\Sigma = (\emptyset, \Sigma_R)$ mit $\Sigma_R = \{(P, 1), (E, 2)\}$

$$\Phi = \left\{ \begin{array}{l} \forall x (P(x) \vee P(y)), \\ \forall x \forall y ((P(x) \wedge E(x, y)) \rightarrow \neg P(y)) \end{array} \right\}$$

- Für $(\mathcal{A}, \alpha)$ mit $\mathcal{A} = (\{0, \dots, 5\}, \llbracket \cdot \rrbracket_{\mathcal{A}})$, wobei $\llbracket P \rrbracket_{\mathcal{A}} = \{0, 2\}$ und $\llbracket E \rrbracket_{\mathcal{A}} = \{(i, i+1) \mid i \in \{0, \dots, 4\}\}$ und $\alpha(x) = 1, \alpha(y) = 2$ gilt $(\mathcal{A}, \alpha) \in \text{Mod}(\Phi)$
- Für $(\mathcal{B}, \beta)$ mit $\mathcal{B} = (\mathbb{Z}, \llbracket \cdot \rrbracket_{\mathcal{B}})$, wobei $\llbracket P \rrbracket_{\mathcal{B}} = 2\mathbb{Z}$ und $\llbracket E \rrbracket_{\mathcal{B}} = \{(i, 2i) \mid i \in \mathbb{Z}\}$ und $\alpha(x) = \alpha(y) = 0$ gilt $(\mathcal{B}, \beta) \notin \text{Mod}(\Phi)$

Für jede endliche Formelmenge $\Phi = \{\varphi_1, \dots, \varphi_n\} \subseteq \text{FOL}(\Sigma, \mathbb{X})$ gilt

$$\text{Mod}(\Phi) = \text{Mod}(\varphi_1 \wedge \dots \wedge \varphi_n)$$

Modelle von Sätzen

Auf den Wert eines Satzes $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$ (Formel ohne freie Variablen) in einer Interpretation $I = (\mathcal{S}, \beta)$ hat die Belegung β keinen Einfluss, d.h. für beliebige Belegungen $\alpha, \beta : \mathbb{X} \rightarrow S$ gilt

$$\llbracket \varphi \rrbracket_{(\mathcal{S}, \alpha)} = \llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)}$$

Beispiel: $\varphi = \forall x \exists y (P(x) \wedge E(x, y) \rightarrow \neg P(y))$

Für Sätze $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$ genügt es also, den Wert in Strukturen (statt in Interpretationen) zu betrachten:

$$\llbracket \varphi \rrbracket_{\mathcal{S}} = \llbracket \varphi \rrbracket_{(\mathcal{S}, \beta)} \quad \text{für jede beliebige Belegung } \beta : \mathbb{X} \rightarrow S$$

Menge aller Modelle des Satzes $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$:

$$\text{Mod}(\varphi) = \{ \mathcal{S} \mid \mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}}) \text{ ist } \Sigma\text{-Struktur und } \llbracket \varphi \rrbracket_{\mathcal{S}} = 1 \}$$

Modelle von Sätzen – Beispiele

$$\varphi = \forall x R(x, x) \wedge \forall x \forall y (R(x, y) \rightarrow R(f(x, y), f(y, x)))$$

mit $\mathbb{X} = \{x, y\}$ und Signatur $\Sigma = (\Sigma_F, \Sigma_R)$, wobei
 $\Sigma_F = \{(f, 2)\}$ und $\Sigma_R = \{(R, 2)\}$

- ▶ Für Σ -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit $A = \{\heartsuit\}$ und
 $\llbracket f \rrbracket_{\mathcal{A}}(\heartsuit, \heartsuit) = \heartsuit$, $\llbracket R \rrbracket_{\mathcal{A}} = \{(\heartsuit, \heartsuit)\}$
gilt $\mathcal{A} \in \text{Mod}(\varphi)$
- ▶ Für Σ -Struktur $\mathcal{B} = (B, \llbracket \cdot \rrbracket_{\mathcal{B}})$ mit $B = \mathbb{N}$ und
 $\forall (m, n) \in \mathbb{N}^2 : \llbracket f \rrbracket_{\mathcal{B}}(m, n) = m$, $\llbracket R \rrbracket_{\mathcal{B}} = \leq$
gilt $\mathcal{B} \in \text{Mod}(\varphi)$
- ▶ Für Σ -Struktur $\mathcal{C} = (C, \llbracket \cdot \rrbracket_{\mathcal{C}})$ mit $C = \{a, b\}^*$ und
 $\llbracket f \rrbracket_{\mathcal{C}} = \circ$, $\llbracket R \rrbracket_{\mathcal{C}} = \sqsubseteq$
gilt $\mathcal{C} \notin \text{Mod}(\varphi)$

Modelle von Sätzen – Beispiele

$$\varphi = \forall x (\neg G(x) \rightarrow G(f(x)))$$

- ▶ $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit A = Menge aller Menschen,
 $\llbracket f \rrbracket_{\mathcal{A}}(x) = \text{Vater von } x$, $\llbracket G \rrbracket_{\mathcal{A}}(x)$ gdw. x männlich $\mathcal{A} \in \text{Mod}(\varphi)$
- ▶ $\mathcal{B} = (B, \llbracket \cdot \rrbracket_{\mathcal{B}})$ mit B = Menge aller Orte 51° nördlicher Breite
 $\llbracket f \rrbracket_{\mathcal{B}}(x) = \text{Ort 50 km westlich von } x$, $\llbracket G \rrbracket_{\mathcal{B}}(x)$ gdw. x in Sachsen
 $\mathcal{B} \notin \text{Mod}(\varphi)$
- ▶ $\mathcal{C} = (\mathbb{N}, \llbracket \cdot \rrbracket_{\mathcal{C}})$ mit
 $\llbracket f \rrbracket_{\mathcal{C}}(x) = x + 1$, $\llbracket G \rrbracket_{\mathcal{C}}(x)$ gdw. x Primzahl $\mathcal{C} \notin \text{Mod}(\varphi)$
- ▶ $\mathcal{D} = (\mathbb{Z}, \llbracket \cdot \rrbracket_{\mathcal{D}})$ mit $\llbracket f \rrbracket_{\mathcal{D}}(x) = x + 1$,
 $\llbracket G \rrbracket_{\mathcal{D}} = 2\mathbb{Z}$ $\mathcal{D} \in \text{Mod}(\varphi)$
- ▶ $\mathcal{E} = (E, \llbracket \cdot \rrbracket_{\mathcal{E}})$ mit $E = \mathbb{R}^2$ (Menge aller Punkte der Ebene),
 $\llbracket f \rrbracket_{\mathcal{E}}(x, y) = (-x, -y)$, $\llbracket G \rrbracket_{\mathcal{E}}(x, y)$ gdw. $x < y$ $\mathcal{E} \notin \text{Mod}(\varphi)$

Modelle von Satzungen – Beispiel

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit $\Sigma_F = \{(f, 2), (e, 0)\}$, $\Sigma_R = \{(\equiv, 2)\}$

$$\Phi = \left\{ \begin{array}{l} \forall x \forall y \forall z (f(f(x, y), z) = f(x, f(y, z))), \\ \forall x (f(x, e) = x) \end{array} \right\}$$

- ▶ Für Σ -Struktur $\mathcal{A} = (\mathbb{N}, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit
 $\llbracket \equiv \rrbracket_{\mathcal{A}} = I_{\mathbb{N}}$, $\forall (m, n) \in \mathbb{N}^2 : \llbracket f \rrbracket_{\mathcal{A}}(m, n) = m \cdot n$, $\llbracket e \rrbracket_{\mathcal{A}} = 1$ gilt
 $\mathcal{A} \in \text{Mod}(\Phi)$
- ▶ Für Σ -Struktur $\mathcal{B} = (\{0, 1\}, \llbracket \cdot \rrbracket_{\mathcal{B}})$ mit
 $\llbracket \equiv \rrbracket_{\mathcal{B}} = I_{\{0, 1\}}$, $\llbracket f \rrbracket_{\mathcal{B}} = \max$, $\llbracket e \rrbracket_{\mathcal{B}} = 1$ gilt $\mathcal{B} \notin \text{Mod}(\Phi)$

$\text{Mod}(\Phi)$ = Menge aller Σ -Strukturen $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}})$, für die
 $\llbracket \equiv \rrbracket_{\mathcal{S}} = I_S$ (Festlegung) und
 S mit $\llbracket f \rrbracket_{\mathcal{S}}$ ein Monoid mit neutralem Element $\llbracket e \rrbracket_{\mathcal{S}}$ bildet

Charakterisierung von Eigenschaften von Strukturen

Festlegung zur Interpretation des zweistelligen Relationssymbols $=$:

In jeder Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}})$ gilt $\llbracket = \rrbracket_{\mathcal{S}} = I_S = \{(x, x) \mid x \in S\}$.

Sätze zur Charakterisierung von Strukturen beschränkter

Mächtigkeit der Trägermengen

► $\varphi_1 = \forall x \forall y (x = y)$

Für jede Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}}) \in \text{Mod}(\varphi_1)$ gilt $|S| \leq 1$

► $\varphi_2 = \exists x \exists y (\neg(x = y))$

Für jede Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}}) \in \text{Mod}(\varphi_2)$ gilt $|S| \geq 2$

► $\varphi_3 = \exists x P(x) \wedge \exists x \neg P(x)$

Für jede Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}}) \in \text{Mod}(\varphi_3)$ gilt $|S| \geq 2$

► $\varphi_4 = \exists x \exists y (R(x, y) \wedge \neg R(y, x))$

Für jede Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}}) \in \text{Mod}(\varphi_4)$ gilt $|S| \geq 2$

► $\varphi_5 = \forall x \forall y \forall z (x = y \vee x = z \vee y = z)$

Für jede Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}}) \in \text{Mod}(\varphi_5)$ gilt $|S| \leq 2$

► $\varphi_6 = \varphi_2 \wedge \varphi_5$

Für jede Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}}) \in \text{Mod}(\varphi_6)$ gilt $|S| = 2$

► Ist $\varphi_7 = \varphi_1 \wedge \varphi_2$ erfüllbar?

Modelle von Satzmengen – Beispiel

Signatur $\Sigma = (\emptyset, \Sigma_R)$ mit $\Sigma_R = \{(R, 2), (P, 1)\}$

$$\Phi = \left\{ \begin{array}{l} \exists x P(x), \\ \exists x \neg P(x), \\ \forall x R(x, x), \\ \forall x \forall y (R(x, y) \rightarrow R(y, x)), \\ \forall x \forall y \forall z ((R(x, y) \wedge R(y, z)) \rightarrow R(x, z)) \end{array} \right\}$$

- ▶ Für Σ -Struktur $\mathcal{A} = (\{a, b, c\}, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit $\llbracket P \rrbracket_{\mathcal{A}} = \{a, c\}$ und $\llbracket R \rrbracket_{\mathcal{A}} = \{(a, a), (a, c), (b, b), (c, a), (c, c)\}$ gilt $\mathcal{A} \in \text{Mod}(\Phi)$
- ▶ Für Σ -Struktur $\mathcal{B} = (\mathbb{N}, \llbracket \cdot \rrbracket_{\mathcal{B}})$ mit $\llbracket P \rrbracket_{\mathcal{B}} = 3\mathbb{N}$ und $\llbracket R \rrbracket_{\mathcal{B}} = \leq$ gilt $\mathcal{B} \notin \text{Mod}(\Phi)$

$\text{Mod}(\Phi)$ = Menge aller Σ -Strukturen $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}})$, für die $\llbracket P \rrbracket_{\mathcal{S}} \neq \emptyset$, $S \setminus \llbracket P \rrbracket_{\mathcal{S}} \neq \emptyset$ (und damit $|S| \geq 2$) und $\llbracket R \rrbracket_{\mathcal{S}}$ eine Äquivalenzrelation ist

Charakterisierung von Strukturen durch Satzmengen

Beispiel: Graphen mit Eckenfärbung mit Farben aus $\{R, G, B\}$
Signatur $\Sigma = (\emptyset, \Sigma_R)$ mit $\Sigma_R = \{(R, 1), (G, 1), (B, 1), (E, 2)\}$

$$\Phi = \left\{ \begin{array}{l} \forall x \neg E(x, x), \\ \forall x \forall y (E(x, y) \rightarrow E(y, x)), \\ \forall x (R(x) \vee G(x) \vee B(x)), \\ \forall x (R(x) \rightarrow \neg (G(x) \vee B(x))), \\ \forall x (G(x) \rightarrow \neg (R(x) \vee B(x))), \\ \forall x (B(x) \rightarrow \neg (G(x) \vee R(x))) \end{array} \right\}$$

(jede Ecke mit genau einer Farbe aus $\{R, G, B\}$ gefärbt)

$\text{Mod}(\Phi)$ = Menge aller ungerichteten schlingenfreien Graphen mit Eckenfärbung mit Farben aus $\{R, G, B\}$

Konfliktfreiheit der Eckenfärbung:

$$\psi = \forall x \forall y (((R(x) \wedge R(y)) \vee (G(x) \wedge G(y)) \vee (B(x) \wedge B(y))) \rightarrow \neg E(x, y))$$

$\text{Mod}(\Phi \cup \{\psi\})$ = Menge aller ungerichteten schlingenfreien Graphen mit konfliktfreier Eckenfärbung mit Farben aus $\{R, G, B\}$

Was bisher geschah

Modellierung von

Eigenschaften (Bedingungen, Anforderungen) an Individuen und Beziehungen durch Formel(menge)n in (klassischen) **Logiken**:

$\text{FOL}(\Sigma, \mathbb{X})$ Prädikatenlogik

- ▶ Syntax
- ▶ Semantik von Formeln, Sätzen, Formelmengen
- ▶ Modellmengen
- ▶ allgemeingültig, erfüllbar, unerfüllbar
- ▶ Äquivalenz

Charakterisierung algebraischer Strukturen durch Satzmenngen

Aussagenlogik als FOL-Spezialfall (Fragment)

Wiederholung $AL(P)$:

Syntax : für Signatur $\Sigma_P = (\Sigma_F, \Sigma_R)$ mit $\Sigma_F = \emptyset$ und $\Sigma_R = \{(p, 0) \mid p \in P\}$. gilt $AL(P) = FOL(\Sigma_P, \emptyset)$
(Alle Formeln in $FOL(\Sigma_P, \emptyset)$ sind Sätze, enthalten keine Individuenvariablen, Terme und Quantoren)

Semantik : Jede Σ_P -Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}})$ definiert für jedes $(p, 0) \in \Sigma_R$ (also für jedes $p \in P$) einen Wahrheitswert $\llbracket p \rrbracket_{\mathcal{S}} \in \{0, 1\}$,
also Funktion $W_{\mathcal{S}} : P \rightarrow \{0, 1\}$ (Belegung aller **Aussagenvariablen** $p \in P$ mit Wahrheitswerten)
mit $\forall p \in P : W_{\mathcal{S}}(p) = \llbracket p \rrbracket_{\mathcal{S}}$

Damit gilt auch für alle Formeln $\varphi \in FOL(\Sigma_P, \emptyset)$:

$$W_{\mathcal{S}}(\varphi) = \llbracket \varphi \rrbracket_{\mathcal{S}}$$

Beispiel

$P = \{p, q, r\}$ definiert die Signatur $\Sigma_{\{p,q,r\}} = (\Sigma_F, \Sigma_R)$
mit $\Sigma_F = \emptyset$ und $\Sigma_R = \{(p, 0), (q, 0), (r, 0)\}$

$$\varphi = p \wedge (\neg r \vee q) \quad \in \quad \text{FOL}(\Sigma_{\{p,q,r\}}, \emptyset) = \text{AL}(\{p, q, r\})$$

$\Sigma_{\{p,q,r\}}$ -Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}})$ mit

$$\llbracket p \rrbracket_{\mathcal{S}} = \llbracket r \rrbracket_{\mathcal{S}} = 1 \quad \text{und} \quad \llbracket q \rrbracket_{\mathcal{S}} = 0$$

entspricht der Belegung (der Aussagenvariablen)

$W_{\mathcal{S}} : \{p, q, r\} \rightarrow \{0, 1\}$:

$$W_{\mathcal{S}} = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$$

Wert von φ in $\mathcal{S}$: $\llbracket \varphi \rrbracket_{\mathcal{S}} = 0 = W_{\mathcal{S}}(\varphi)$

Modellmenge von φ :

- ▶ in $\text{AL}(\{p, q, r\})$: $\text{Mod}(\varphi) = \{W_{100}, W_{110}, W_{111}\}$
- ▶ in $\text{FOL}(\Sigma_{\{p,q,r\}}, \emptyset)$: $\text{Mod}(\varphi) = \{(\mathcal{S}, \beta) \mid \Sigma_{\{p,q,r\}}\text{-Struktur } \mathcal{S} \text{ mit } (\llbracket p \rrbracket_{\mathcal{S}}, \llbracket q \rrbracket_{\mathcal{S}}, \llbracket r \rrbracket_{\mathcal{S}}) \in \{(1, 0, 0), (1, 1, 0), (1, 1, 1)\}\}$

Semantisches Folgern in FOL

(analog Aussagenlogik)

Für $\Phi \subseteq \text{FOL}(\Sigma, \mathbb{X})$ (Kontext) und $\psi \in \text{FOL}(\Sigma, \mathbb{X})$ (Aussage) gilt

$$\Phi \models \psi \quad \text{gdw.} \quad \text{Mod}(\Phi) \subseteq \text{Mod}(\psi)$$

(Behauptung ψ ist **semantische Folgerung** aus dem Kontext Φ .)

$$\text{Folgerungsrelation} \quad \models \subseteq 2^{\text{FOL}(\Sigma, \mathbb{X})} \times \text{FOL}(\Sigma, \mathbb{X})$$

Beispiele:

- ▶ $\{p, p \rightarrow q\} \models q$
- ▶ $\{P(a), P(a) \rightarrow Q(a)\} \models Q(a)$
- ▶ $\{\forall x(P(x) \vee Q(x)), \neg P(y)\} \models Q(y)$
- ▶ $\{\forall x(P(x) \vee Q(x)), \neg P(x)\} \models Q(x)$
- ▶ $\models Q(a) \vee \neg Q(a)$

Semantisches Folgern – Beispiele

- ▶ für $\Phi = \{\forall x(P(x) \vee Q(x)), \neg P(x)\}$ und $\psi = Q(y)$ gilt $\Phi \not\models \psi$, weil
z.B. für $(\mathcal{S}, \beta)$ mit $\mathcal{S} = (\{a, b\}, \llbracket \cdot \rrbracket_{\mathcal{S}})$ mit
 $\llbracket P \rrbracket_{\mathcal{S}} = \{b\}$, $\llbracket Q \rrbracket_{\mathcal{S}} = \{a\}$ und $\beta = \{x \mapsto a, y \mapsto b\}$ gilt:
 $(\mathcal{S}, \beta) \in \text{Mod}(\Phi)$, aber $(\mathcal{S}, \beta) \notin \text{Mod}(\psi)$
- ▶ $\{\forall x (P(x) \vee Q(x)), \forall x \neg P(x)\} \models \forall x Q(x)$
- ▶ für $\Phi' = \{\forall x (P(x) \vee Q(x)), \exists x \neg P(x)\}$ und $\psi' = (\forall x Q(x))$ gilt $\Phi' \not\models \psi'$, weil
z.B. für $\mathcal{S} = (\{a, b\}, \llbracket \cdot \rrbracket_{\mathcal{S}})$ mit $\llbracket P \rrbracket_{\mathcal{S}} = \{b\}$ und $\llbracket Q \rrbracket_{\mathcal{S}} = \{a\}$ gilt
 $\mathcal{S} \in \text{Mod}(\Phi')$, aber $\mathcal{S} \notin \text{Mod}(\psi')$

Für alle Formeln $\varphi, \psi \in \text{FOL}(\Sigma, \mathbb{X})$ gilt

- ▶ $\{\varphi \rightarrow \psi, \varphi\} \models \psi$ (Modus ponens)
- ▶ $\{\varphi \rightarrow \psi, \neg\psi\} \models \neg\varphi$ (Modus tollens)
- ▶ $\forall x \varphi \models \varphi$

Beispiel: Formale Beschreibung und Folgern

Beispiel: Jeder (ungerichtete, schlingenfreie) Graph mit (wenigstens) einer Kante hat (wenigstens) zwei Knoten.

Kontext

informal: (ungerichteter, schlingenfreier) Graph (V, E)
(Menge V mit zweistelliger Kantenrelation E)
mit (wenigstens) einer Kante

Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit $\Sigma_F = \emptyset$ und $\Sigma_R = \{(E, 2), (=, 2)\}$

Axiome (Formelmenge, die genau alle ungerichteten schlingenfreien Graphen mit wenigstens einer Kante erfüllen):

$$\Phi = \{\forall x \forall y (E(x, y) \rightarrow E(y, x)), \forall x \neg E(x, x), \exists x \exists y E(x, y)\}$$

Behauptung

informal : Der Graph hat (wenigstens) zwei Knoten.

Formel $\psi = \exists x \exists y \neg (x = y)$

Frage: $\Phi \stackrel{?}{\models} \psi$ (Folgt ψ aus Φ ?)

(wird oft gezeigt durch die äquivalente Aussage: $\Phi \cup \{\neg \psi\}$ unerfüllbar)
keine allgemeine (maschinelle) Lösung in FOL möglich

Logisches Schließen in der Prädikatenlogik

Aussagen wie z.B. die folgende lassen sich i.A. **nicht** durch Nachprüfen aller Interpretationen zeigen:

Beispiel:

In jeder Äquivalenzrelation gilt

$$\psi = \forall x \forall y \forall z (R(x, y) \wedge R(x, z) \rightarrow R(y, z))$$

d. h.

$$\left\{ \begin{array}{l} \forall x R(x, x) \\ \forall x \forall y (R(x, y) \rightarrow R(y, x)) \\ \forall x \forall y \forall z (R(x, y) \wedge R(y, z) \rightarrow R(x, z)) \end{array} \right\} \models \psi$$

Satz: Nachprüfen in Σ -Termstrukturen als allgemeinsten Σ -Strukturen genügt.

(syntaktische Interpretationen)

Ableitbarkeit in Kalkülen (syntaktisches Schließen) ist daher in der Prädikatenlogik noch wichtiger als in der Aussagenlogik.

Prädikatenlogische Ableitungen

Ableitungsrelation z.B. durch Erweiterung des Resolutionsverfahrens auf Prädikatenlogik $\text{FOL}(\Sigma, \mathbb{X})$

- ▶ Interpretation in der Termalgebra über Σ
- ▶ Hinzufügen neuer Konstanten, falls notwendig
- ▶ Erweiterung um zusätzliche Inferenz-Regeln, z.B.

$$\forall x \varphi \vdash \varphi_{[x \mapsto t]} \quad \text{für beliebigen Grundterm } t$$

$$\varphi(t) \vdash \exists x \varphi_{[t \mapsto x]} \quad \text{für } x \notin \text{fvar}(\varphi)$$

$$\exists x \varphi \vdash \varphi_{[x \mapsto c]} \quad \text{für neue Konstante } c$$

Grundlagen der logischen Programmierung

(mehr dazu im Modul Grundlagen der Künstlichen Intelligenz)

Gültigkeitsproblem

(Allgemein)-Gültigkeitsproblem in einer Logik:

gegeben: Formel φ in dieser Logik

Frage: Ist φ allgemeingültig?

für jede Logik interessant:

Gibt es (wenigstens) ein automatisches Verfahren (Algorithmus),
der dieses Problem **entscheidet**,

also zu **jeder** gegebenen Formel dieser Logik die Frage in endlicher
Zeit korrekt mit ja oder nein beantwortet.

Logiken, für die es solche Algorithmen gibt, heißen **entscheidbar**.

(Maschinelle) Lösbarkeit

Ziel: Bestimmung **aller** allgemeingültigen Formeln $\in \text{FOL}(\Sigma, \mathbb{X})$

Durch Erzeugung aller möglichen Beweise lassen sich alle allgemeingültigen Formeln $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$ nacheinander angeben.

(Klassische **Prädikatenlogik** der ersten Stufe ist **aufzählbar**.)

Es gibt kein Verfahren, welches für jede Formel $\varphi \in \text{FOL}(\Sigma, \mathbb{X})$ in endlicher Zeit bestimmt, ob φ allgemeingültig ist (irgendwann in der Aufzählung vorkommt).

(Klassische **Prädikatenlogik** der ersten Stufe ist **unentscheidbar**.)

(mehr dazu im Modul zur Berechenbarkeit)

Unterschied zur Aussagenlogik:

Für jede Formel $\varphi \in \text{AL}(P)$ läßt sich in endlicher Zeit algorithmisch bestimmen, ob φ allgemeingültig ist.

(Test endlich vieler endlicher Strukturen genügt, z.B. Wahrheitstabellen)

(Klassische **Aussagenlogik** ist **entscheidbar**.)

Eignung von Logiken zur Modellierung

abhängig vom Aufgabenbereich

Verschiedene Logiken unterscheiden sich in

- ▶ **Ausdrucksstärke:**
 - ▶ Aussagen über welche Bereiche lassen sich überhaupt formulieren?
 - ▶ kompakte Darstellung typischer Aussagen
 - ▶ Verständlichkeit der Formalisierung

Eignung zur **Repräsentation** von Informationen

- ▶ algorithmische Entscheidbarkeit wichtiger Fragen
(z.B. Erfüllbarkeit, Allgemeingültigkeit, Folgerungen)
Eignung zur (maschinellen) **Verarbeitung** von Informationen

Vergleich verschiedener Logiken

klassische Logiken:

- ▶ klassische Prädikatenlogik der ersten Stufe FOL:
 - ▶ hohe Ausdrucksstärke,
 - ▶ kompakte, verständliche Formulierungen,
 - ▶ unentscheidbar
- ▶ klassische Aussagenlogik AL (Spezialfall, Fragment von FOL):
 - ▶ geringe Ausdrucksstärke
 - ▶ umfangreiche Darstellung
 - ▶ entscheidbar
 - ▶ SAT-Solver zur maschinellen Lösung

nichtklassische Logiken: Speziallogiken für verschiedene Anwendungen

- ▶ Temporallogiken (Spezifikation, Verifikation von Systemen)
- ▶ Hoare-Logik (Spezifikation, Verifikation imperativer Software)
- ▶ Beschreibungslogiken (Wissensextraktion aus Ontologien, Semantic Web)

oft entscheidbare Fragmente von FOL mit modifizierter Syntax
(mehr dazu in verschiedenen Modulen später im Studium)

Mehrsortige Strukturen

Modellierung von Strukturen mit verschiedenen **Sorten**
(Trägmengen, Typen) $\mathbb{S} = \{S_i \mid i \in I\}$ von Elementen

Beispiele mehrsortiger Strukturen:

- ▶ Sorten: Personen, Länder
Relation: $H \subseteq \text{Personen} \times \text{Länder}$
wobei $(p, l) \in H$ gdw. p kommt aus l
- ▶ Sorten: Personen, Bücher
Relation: $A \subseteq \text{Personen} \times \text{Bücher}$
wobei $(p, b) \in A$ gdw. p ist Autor von b
- ▶ Sorten: Personen, $\mathbb{N}$, Orte
Relation: $G \subseteq \text{Personen} \times \mathbb{N} \times \text{Orte}$
wobei $(p, j, o) \in G$ gdw. p wurde im Jahr j in o geboren

Mehrsortige Signaturen

mehrsortige Signatur $\Sigma = (\mathbb{S}, \Sigma_F, \Sigma_R)$ besteht aus

$\mathbb{S} = \{S_i \mid i \in I\}$: Menge der Symbole für **Sorten** (Typen)

Σ_F : Menge der Funktionssymbole
mit Argument- und Ergebnistypen ($f : \mathbb{S}^* \rightarrow \mathbb{S}$)

Σ_R : Menge der Relationssymbole
mit Argumenttypen ($s \subseteq \mathbb{S}^*$)

Beispiel: Bücher mit Autoren und Erscheinungsjahr

Sorten B (Bücher), P (Personen), $\mathbb{Z}$ (Jahreszahlen),
 2^B (Mengen von Büchern), also $\mathbb{S} = \{B, P, \mathbb{Z}, 2^B\}$

Funktionssymbole (mit Argument- und Ergebnistypen):

$$\Sigma_F = \left\{ \begin{array}{lll} \text{Erscheinungsjahr} & : B & \rightarrow \mathbb{Z} \\ \text{erstes-Buch-von} & : P & \rightarrow B \\ \text{gemeinsame-Buecher-von} & : P^2 & \rightarrow 2^B \end{array} \right\}$$

Relationen (mit Argumenttypen):

$$\Sigma_R = \left\{ \begin{array}{ll} \text{ist-Autor-von} & \subseteq P \times B \\ \text{sind-Coautoren} & \subseteq P^2 \end{array} \right\}$$

Terme und Formeln über mehrsortigen Signaturen

gegeben: mehrsortige Signatur Σ mit Menge $\mathbb{S}$ von Sorten

Variablenmenge $\mathbb{X}' = \mathbb{X} \times \mathbb{S}$

(jede Variable mit einer zugeordneter Sorte aus $\mathbb{S}$)

Definition (induktiv)

Für jede Sorte $S \in \mathbb{S}$ ist die Menge $\text{Term}_S(\Sigma, \mathbb{X}')$ aller

Terme der Sorte S mit Variablen aus der Menge $\mathbb{X}'$ definiert durch:

IA: $\{x \mid (x : S) \in \mathbb{X}'\} \subseteq \text{Term}_S(\Sigma, \mathbb{X}')$

IS: Für alle f mit $(f : \mathbb{S}^* \rightarrow S) \in \Sigma$ und alle Tupel $(t_1, \dots, t_n)$, wobei für alle $k \in \{1, \dots, n\}$ gilt $t_k \in \text{Term}_{S_k}(\Sigma, \mathbb{X}')$,
ist $f(t_1, \dots, t_n) \in \text{Term}_S(\Sigma, \mathbb{X}')$

Spezialfall: Jede Konstante $f : S \in \Sigma$ ist Σ -Term der Sorte S .

Beispiele: $t = \text{Erscheinungsjahr}(\text{erstes-Buch-von}(x))$

mit $(x : P) \in \mathbb{X}'$ ist ein Term in $\text{Term}_J(\Sigma, \mathbb{X}')$

$\text{erstes-Buch-von}(\text{Erscheinungsjahr}(x))$ ist kein Σ -Term

Erweiterung auf **Atome** und **Formeln** analog einsortigen Signaturen

(meist zu quantifizierten Variablen jeweils Angabe der Sorte)

Mehrsortige Σ -Strukturen

gegeben: mehrsortige Signatur $\Sigma = (\mathbb{S}, \Sigma_F, \Sigma_R)$

mehrsortige Σ -Struktur $\mathcal{A} = (\{A_S \mid S \in \mathbb{S}\}, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit

- ▶ für jede Sorte $S \in \mathbb{S}$ eine nichtleere Menge A_S (Trägermenge, Universum der Sorte)
- ▶ für jedes Funktionssymbol $f : S_1 \times \cdots \times S_n \rightarrow S$ eine Funktion $\llbracket f \rrbracket_{\mathcal{A}} : A_{S_1} \times \cdots \times A_{S_n} \rightarrow A_S$
- ▶ für jedes Relationssymbol $R \subseteq S_1 \times \cdots \times S_n$ eine Relation $\llbracket R \rrbracket_{\mathcal{A}} \subseteq A_{S_1} \times \cdots \times A_{S_n}$

Beispiel Vektorraum (WH)

(bekannt aus Modul Mathematik)

mehrsortige algebraische Struktur mit

- ▶ Sorten (Symbole für Trägermengen):
 - ▶ Menge V von Vektoren
 - ▶ Menge S von Skalaren
- ▶ Operationen, z.B.
 - ▶ Addition von Skalaren $+_s : S \times S \rightarrow S$
 - ▶ Multiplikation von Skalaren $\cdot_s : S \times S \rightarrow S$
 - ▶ Addition von Vektoren $+_v : V \times V \rightarrow V$
 - ▶ Multiplikation von Skalaren mit Vektoren $\odot : S \times V \rightarrow V$
 - ▶ Skalarprodukt $\langle \cdot, \cdot \rangle : V \times V \rightarrow S$
 - ▶ Skalare $n_s \in S, e_s \in S$
 - ▶ Nullvektor $n_v \in V$
- ▶ Relationen, z.B.
 - ▶ Gleichheit von Skalaren $=_s \subseteq S^2$
 - ▶ Gleichheit von Vektoren $=_v \subseteq V^2$

Beispiel Vektorraum – Signatur

mehrsortige Signatur $\Sigma_V = (\mathbb{S}, \Sigma_F, \Sigma_R)$ mit

$$\begin{aligned}\mathbb{S} &= \{S, V\} \\ \Sigma_F &= \left\{ \begin{array}{llll} +_S & : & S^2 & \rightarrow S \\ \cdot_S & : & S^2 & \rightarrow S \\ +_V & : & V^2 & \rightarrow V \\ \odot & : & S \times V & \rightarrow V \\ n_S & : & & S \\ e_S & : & & S \\ n_V & : & & V \end{array} \right\} \\ \Sigma_R &= \{=_S \subseteq S^2, =_V \subseteq V^2\}\end{aligned}$$

Vektorraum =

(mehrsortige) Σ_V -Struktur $\mathcal{A} = (\{A_S \mid S \in \mathbb{S}\}, [\cdot]_{\mathcal{A}})$, welche alle für Vektorräume charakteristischen Bedingungen (Axiome) erfüllt.

Beispiel Vektorraum – WH: Definition

Struktur $(S, V, +_s, \cdot_s, \odot, n_v, n_s, e_s)$ ist Vektorraum

gdw. $(S, +_s, \cdot_s, n_s, e_s)$ ist ein Körper, d.h. erfüllt die Axiome

$$\forall x \in S \forall y \in S \forall z \in S \quad ((x +_s y) +_s z =_s x +_s (y +_s z))$$

$$\forall x \in S \forall y \in S \quad (x +_s y =_s y +_s x)$$

$$\forall x \in S \forall y \in S \forall z \in S \quad ((x \cdot_s y) \cdot_s z =_s x \cdot_s (y \cdot_s z))$$

$\vdots$

$$\forall x \in S \forall y \in S \forall z \in S \quad (x \cdot_s (y +_s z) =_s (x \cdot_s y) +_s (x \cdot_s z))$$

und erfüllt außerdem (Axiome für $\odot$ und n_v)

$$\forall x \in S \forall y \in S \forall v \in V \quad ((x \cdot_s y) \odot v =_v x \odot (y \odot v))$$

$$\forall x \in S \forall y \in S \forall v \in V \quad ((x +_s y) \odot v =_v x \odot v +_v y \odot v)$$

$$\forall x \in S \forall u \in V \forall v \in V \quad (x \odot (u +_v v) =_v x \odot u +_v x \odot v)$$

$$\forall v \in V \quad (e_s \odot v =_v v)$$

$$\forall v \in V \quad (n_s \odot v =_v n_v)$$

Axiome: Menge $\Phi_V \in \text{FOL}(\Sigma_V, \mathbb{X})$ (mehrsortiger) Formeln (Sätze) mit
 $\mathbb{X} = \{(x : S), (y : S), (z : S), (u : V), (v : V)\}$

Vektorraum – Beispiel

Vektorraum:

Σ_V -Struktur, die alle Axiome (Formeln aus Φ_V) erfüllt

Beispiel: Σ_V -Struktur $\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit

$$A_S = \{0, 1\} \quad (\text{Menge der Skalare})$$

$$A_V = \{\vec{0}\} \quad (\text{Menge der Vektoren})$$

$$\llbracket n_s \rrbracket_{\mathcal{A}} = 0, \quad \llbracket e_s \rrbracket_{\mathcal{A}} = 1 \quad (\in A_S)$$

$$\llbracket n_v \rrbracket_{\mathcal{A}} = \vec{0} \quad (\in A_V)$$

$$\forall x, y \in A_S : \llbracket +_s \rrbracket_{\mathcal{A}}(x, y) = \begin{cases} 0 & \text{falls } x = y \\ 1 & \text{sonst} \end{cases} \quad (\in A_S)$$

$$\forall x, y \in A_S : \llbracket \cdot_s \rrbracket_{\mathcal{A}}(x, y) = x \cdot y \quad (\in A_S)$$

$$\llbracket \odot \rrbracket_{\mathcal{A}}(0, \vec{0}) = \llbracket \odot \rrbracket_{\mathcal{A}}(1, \vec{0}) = \vec{0} \quad (\in A_V)$$

$$\llbracket =_s \rrbracket_{\mathcal{A}} = I_{(A_S)} = \{(0, 0), (1, 1)\}$$

$$\llbracket =_v \rrbracket_{\mathcal{A}} = I_{(A_V)} = \{(\vec{0}, \vec{0})\}$$

erfüllt alle Sätze in Φ_V (Axiome)

Menge aller Vektorräume : Modellmenge $\text{Mod}(\Phi_V)$

Was bisher geschah

Modellierung von

Eigenschaften (Bedingungen, Anforderungen) an Individuen und Beziehungen durch Formel(menge)n in (klassischen) **Logiken**:

$\text{FOL}(\Sigma, \mathbb{X})$ Prädikatenlogik

- ▶ Syntax
- ▶ Semantik von Formeln, Sätzen, Formelmengen
- ▶ Modellmengen
- ▶ allgemeingültig, erfüllbar, unerfüllbar
- ▶ Äquivalenz

$\text{AL}(P)$ Aussagenlogik als Spezialfall

Charakterisierung algebraischer Strukturen durch Satzmengen

Aussagenlogik als FOL-Spezialfall (Fragment)

Wiederholung $AL(P)$:

Syntax : für Signatur $\Sigma_P = (\Sigma_F, \Sigma_R)$ mit $\Sigma_F = \emptyset$ und $\Sigma_R = \{(p, 0) \mid p \in P\}$. gilt $AL(P) = FOL(\Sigma_P, \emptyset)$
(Alle Formeln in $FOL(\Sigma_P, \emptyset)$ sind Sätze, enthalten keine Individuenvariablen, Terme und Quantoren)

Semantik : Jede Σ_P -Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}})$ definiert für jedes $(p, 0) \in \Sigma_R$ (also für jedes $p \in P$) einen Wahrheitswert $\llbracket p \rrbracket_{\mathcal{S}} \in \{0, 1\}$,
also Funktion $W_{\mathcal{S}} : P \rightarrow \{0, 1\}$ (Belegung aller **Aussagenvariablen** $p \in P$ mit Wahrheitswerten)
mit $\forall p \in P : W_{\mathcal{S}}(p) = \llbracket p \rrbracket_{\mathcal{S}}$

Damit gilt auch für alle Formeln $\varphi \in FOL(\Sigma_P, \emptyset)$:

$$W_{\mathcal{S}}(\varphi) = \llbracket \varphi \rrbracket_{\mathcal{S}}$$

Beispiel

$P = \{p, q, r\}$ definiert die Signatur $\Sigma_{\{p,q,r\}} = (\Sigma_F, \Sigma_R)$
mit $\Sigma_F = \emptyset$ und $\Sigma_R = \{(p, 0), (q, 0), (r, 0)\}$

$$\varphi = p \wedge (\neg r \vee q) \quad \in \quad \text{FOL}(\Sigma_{\{p,q,r\}}, \emptyset) = \text{AL}(\{p, q, r\})$$

$\Sigma_{\{p,q,r\}}$ -Struktur $\mathcal{S} = (S, \llbracket \cdot \rrbracket_{\mathcal{S}})$ mit

$$\llbracket p \rrbracket_{\mathcal{S}} = \llbracket r \rrbracket_{\mathcal{S}} = 1 \quad \text{und} \quad \llbracket q \rrbracket_{\mathcal{S}} = 0$$

entspricht der Belegung (der Aussagenvariablen)

$W_{\mathcal{S}} : \{p, q, r\} \rightarrow \{0, 1\}$:

$$W_{\mathcal{S}} = \{p \mapsto 1, q \mapsto 0, r \mapsto 1\}$$

Wert von φ in $\mathcal{S}$: $\llbracket \varphi \rrbracket_{\mathcal{S}} = 0 = W_{\mathcal{S}}(\varphi)$

Modellmenge von φ :

- ▶ in $\text{AL}(\{p, q, r\})$: $\text{Mod}(\varphi) = \{W_{100}, W_{110}, W_{111}\}$
- ▶ in $\text{FOL}(\Sigma_{\{p,q,r\}}, \emptyset)$: $\text{Mod}(\varphi) = \{(\mathcal{S}, \beta) \mid \Sigma_{\{p,q,r\}}\text{-Struktur } \mathcal{S} \text{ mit } (\llbracket p \rrbracket_{\mathcal{S}}, \llbracket q \rrbracket_{\mathcal{S}}, \llbracket r \rrbracket_{\mathcal{S}}) \in \{(1, 0, 0), (1, 1, 0), (1, 1, 1)\}\}$

Was bisher geschah

Mehrwertige

Signaturen Symbole für Sorten (Typen)
Funktionssymbolen sind (statt Stelligkeit) Sorten für Argumente und Ergebnis zugeordnet
Relationssymbolen sind (statt Stelligkeit) Sorten für Argumente zugeordnet

Strukturen zu jedem Sortensymbol eine Trägemenge
Interpretation von Funktions- und
Relationenssymbolen durch Funktionen und
Relationen passender Sorten

Modellierung von

Bedingungen (Eigenschaften, Anforderungen) an Individuen und
Beziehungen durch Formel(menge)n in (klassischen)
Logiken:

$\text{FOL}(\Sigma, \mathbb{X})$ Prädikatenlogik mit

$\text{AL}(P)$ Aussagenlogik (Fragment, Spezialfall)

Charakterisierung von Strukturen als Modelle von Satzmengen

Relationen als mehrsortige Funktionen (zweistellig)

Idee: Ersetzung von $\llbracket R \rrbracket_{\mathcal{A}} \subseteq A^2$ durch $\chi_{\llbracket R \rrbracket_{\mathcal{A}}} : A^2 \rightarrow \mathbb{B}$
bisher z.B.

- ▶ (einsortige) Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit $\Sigma_F = \emptyset$ und $\Sigma_R = \{(\textcolor{red}{R}, 2)\}$
- ▶ Interpretation von R in (einsortigen) Σ -Strukturen $\mathcal{A} = (A, \llbracket R \rrbracket_{\mathcal{A}})$ mit $\llbracket R \rrbracket_{\mathcal{A}} \subseteq A \times A$

jetzt: Einführung der Sorten

S (für Trägermenge) und

$\mathbb{B} = \{0, 1\}$ (für Wahrheitswerte)

- ▶ $\{S, \mathbb{B}\}$ -sortige Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ mit $\Sigma_F = \{f_R : S \times S \rightarrow \mathbb{B}\}$ und $\Sigma_R = \emptyset$
- ▶ Interpretation in $\{S, \mathbb{B}\}$ -sortigen Σ -Strukturen $\mathcal{A}' = (\{A, \{0, 1\}\}, \llbracket \cdot \rrbracket_{\mathcal{A}'})$ mit $\llbracket f_R \rrbracket_{\mathcal{A}'} : A \times A \rightarrow \{0, 1\}$ mit

$$\forall (x, y) \in A^2 : \llbracket f_R \rrbracket_{\mathcal{A}'}(x, y) = \chi_{\llbracket R \rrbracket_{\mathcal{A}}}(x, y)$$

$$\text{d.h. } \llbracket f_R \rrbracket_{\mathcal{A}'}(x, y) = 1 \text{ gdw. } (x, y) \in \llbracket R \rrbracket_{\mathcal{A}}$$

Relationen als mehrsortige Funktionen (mehrstellig)

für beliebige Stelligkeit n :

Übersetzung jedes Relationssymbols $(R, n) \in \Sigma_R$ in ein

Funktionssymbol $f_R : S^n \rightarrow \mathbb{B}$, so dass

in jeder $\{S, \mathbb{B}\}$ -sortigen Σ -Struktur $\mathcal{A}' = (\{A, \{0, 1\}\}, \llbracket \cdot \rrbracket_{\mathcal{A}'})$ gilt:

$\llbracket f_R \rrbracket_{\mathcal{A}'} : A^n \rightarrow \{0, 1\}$ mit $\llbracket f_R \rrbracket_{\mathcal{A}'} = \chi_{\llbracket R \rrbracket_{\mathcal{A}}}$, d.h.

$\forall (x_1, \dots, x_n) \in A^n : \llbracket f_R \rrbracket_{\mathcal{A}'}(x_1, \dots, x_n) = 1 \text{ gdw. } (x_1, \dots, x_n) \in \llbracket R \rrbracket_{\mathcal{A}}$

Jeder (einsortigen) Signatur $\Sigma = (\Sigma_F, \Sigma_R)$ lässt sich so eine $\{S, \mathbb{B}\}$ -sortige rein funktionale Signatur $\Sigma' = (\Sigma'_F \cup \Sigma'_R, \emptyset)$ zuordnen:

$$\Sigma'_F = \{f : S^n \rightarrow S \mid (f, n) \in \Sigma_F\}$$

$$\Sigma'_R = \{f_R : S^n \rightarrow \mathbb{B} \mid (R, n) \in \Sigma_R\}$$

Analog lassen sich mehrsortige Relationen durch mehrsortige Funktionen ersetzen.

Es genügt also, **mehrsortige Strukturen ohne Relationen** (Algebren) zu betrachten.

Mehrsortige Strukturen in der Informatik

Programmierung:

Datentypen (Sorten) `int`, `float`, `bool`, `string`
mit Operationen (Signatur), z.B.:

<code>floor</code>	<code>: float</code>	<code>→ int</code>
<code>duplicate</code>	<code>: string × int</code>	<code>→ string</code>
<code>length</code>	<code>: string</code>	<code>→ int</code>
<code>></code>	<code>: float × float</code>	<code>→ bool</code>

Datenbanken:

Tabellen sind extensionale Darstellungen (meist) mehrsortiger
Relationen

Charakterisierung von mehrsortigen Strukturen

Mengen von (algebraischen) Strukturen sind eindeutig definiert durch

Syntax:

Trägmengen (Typen, Sorten), z.B. $\{0, 1\}, \mathbb{N}, \{a, b\}^*, 2^M$

Funktionen (Operationen), z.B. $\text{len} : \{a, b\}^* \rightarrow \mathbb{N}, + : \mathbb{N}^2 \rightarrow \mathbb{N},$
 $\circ : (\{a, b\}^*)^2 \rightarrow \{a, b\}^*, 0 \in \mathbb{N}, \varepsilon \in \{a, b\}^*, \emptyset \in 2^M$

Relationen , z.B. $\leq \subseteq \mathbb{N}^2, \text{prim} \subseteq \mathbb{N}, \sqsubseteq \subseteq (\{a, b\}^*)^2,$

Semantik:

Zusammenhänge zwischen Funktionen und Relationen, z.B.

$$\forall (x, y, z) \in \mathbb{N}^3 \quad ((x + y) + z = x + (y + z))$$

$$\forall x \in \mathbb{N} \quad (x + 0 = x)$$

$$\forall (x, y) \in \mathbb{N}^2 \quad (x + y = y + x)$$

$$\forall u \in \{a, b\}^* \quad : \quad (\text{len}(u) = 0) \leftrightarrow (u = \varepsilon)$$

$$\forall (u, v) \in (\{a, b\}^*)^2 \quad : \quad \text{len}(u \circ v) = \text{len}(u) + \text{len}(v)$$

$$\forall (u, v) \in (\{a, b\}^*)^2 \quad : \quad (u \sqsubseteq v) \rightarrow (\text{len}(u) \leq \text{len}(v))$$

Formale Darstellung von (algebraischen) Strukturen

(abstrakte Algebra, **abstrakter Datentyp**, ADT)

- ▶ **Signatur** $\Sigma = (\mathbb{S}, \Sigma_F, \Sigma_R)$ mit
 - ▶ Menge $\mathbb{S}$ von **Sortensymbolen**,
 - ▶ Menge Σ_F von **Funktionssymbolen** mit Stelligkeit / Typdeklaration
 - ▶ Menge Σ_R von **Relationssymbolen** mit Stelligkeit / Typdeklaration
- ▶ Menge von **Axiomen** beschreibt Zusammenhänge zwischen den Symbolen
(meist prädikatenlogische Formeln, generalisierte Gleichungen)

ADT zur Definition von Software-Schnittstellen:
Festlegung der

Syntax in Sorten und Signatur

(z.B. Java-Interfaces, abstrakte Klassen in C++ ähnlich)

Semantik (Anforderungen an Implementierung) in Axiomen

Abstraktion von konkreten Repräsentationen der Daten (Trägersmengen)
und konkrete Implementierung der Operationen

Algebraische Spezifikation

abstrakte Datentypen (ADT) werden definiert durch

Syntax : (mehrsortige) Signatur

$$\Sigma = (\mathbb{S}, \Sigma_F, \Sigma_R),$$

Semantik : Menge von Axiomen (Sätzen)

$$\Phi \subseteq \text{FOL}(\Sigma, \mathbb{X})$$

zur Modellierung zusammenhängender Datenbereiche
und derer Eigenschaften

konkrete Datentypen : Σ -Strukturen (Syntax)
in $\text{Mod}(\Phi)$ (Semantik)

Abstrakter Datentyp (Spezifikation, Schnittstellen-Beschreibung)
 (Σ, Φ)

repräsentiert eine

Menge

konkreter Datentypen (Σ -Strukturen, Implementierungen)

$\text{Mod}(\Phi)$

Modellierungsbeispiel Navigation

ADT für (Blickrichtung in)

Himmelsrichtungen Norden, Osten, Süden, Westen

mit Operationen für Rechts- und Linksabbiegen und Umlenken

ADT Himmelsrichtungen (Spezifikation):

Sorten: eine Sorte (Himmelsrichtung) $HR = \{N, O, S, W\}$

Signatur:

$N, O, S, W : \quad \quad \quad HR$

$\text{rechts, links, um} : \quad HR \rightarrow \quad HR$

Axiome:

$$\left\{ \begin{array}{l} \text{rechts}(N) = O, \text{links}(N) = W, \dots \\ \text{um}(N) = S, \text{um}(O) = W, \dots \\ \forall h \in HR (\text{rechts}(\text{links}(h)) = h), \\ \forall h \in HR (\text{links}(\text{rechts}(h)) = h), \\ \forall h \in HR (\text{rechts}(\text{rechts}(h)) = \text{um}(h)), \dots \end{array} \right\}$$

Konkrete Datentypen (Beispiele)

Implementierung des ADT Himmelsrichtungen, z.B.:

- als Punkte im $\mathbb{R}^2$ mit Drehmatrizen $\begin{pmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{pmatrix}$

$$\varphi = -\pi/2 : R = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad \varphi = \pi/2 : L = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

$\mathcal{A} = (A, \llbracket \cdot \rrbracket_{\mathcal{A}})$ mit $A = \mathbb{R}^2$ und

$\llbracket N \rrbracket_{\mathcal{A}} = (0, 1)$, $\llbracket O \rrbracket_{\mathcal{A}} = (1, 0)$, $\llbracket S \rrbracket_{\mathcal{A}} = (0, -1)$, $\llbracket W \rrbracket_{\mathcal{A}} = (-1, 0)$,

$$\forall (x, y) \in \mathbb{R}^2 \left(\llbracket \text{rechts} \rrbracket_{\mathcal{A}}(x, y) = R \begin{pmatrix} x \\ y \end{pmatrix} \wedge \llbracket \text{links} \rrbracket_{\mathcal{A}}(x, y) = L \begin{pmatrix} x \\ y \end{pmatrix} \right)$$

$$\forall (x, y) \in \mathbb{R}^2 (\llbracket \text{um} \rrbracket_{\mathcal{A}}(x, y) = (-x, -y))$$

Ist diese Implementierung korrekt (d.h. Modell aller Axiome)? (ÜA)

- $\mathcal{B} = (B, \llbracket \cdot \rrbracket_{\mathcal{B}})$ mit $B = \{0, 1, 2, 3\}$ und
 $\llbracket N \rrbracket_{\mathcal{B}} = 0$, $\llbracket O \rrbracket_{\mathcal{B}} = 1$, $\llbracket S \rrbracket_{\mathcal{B}} = 2$, $\llbracket W \rrbracket_{\mathcal{B}} = 3$,
 $\forall x \in \{0, 1, 2, 3\} (\llbracket \text{rechts} \rrbracket_{\mathcal{B}}(x) = (x + 1) \bmod 4)$
 $\forall x \in \{0, 1, 2, 3\} (\llbracket \text{links} \rrbracket_{\mathcal{B}}(x) = (x - 1) \bmod 4)$,
 $\forall x \in \{0, 1, 2, 3\} (\llbracket \text{um} \rrbracket_{\mathcal{B}}(x) = (x + 2) \bmod 4)$
Ist diese Implementierung korrekt?

(ÜA)

Modellierungsbeispiel ID-Nummern

Ziel: **ADT Menge** zur Verwaltung (endlicher) Mengen natürlicher Zahlen (z.B. ID-Nummern), so dass festgestellt werden kann, ob Zahlen in Mengen enthalten sind, zu Mengen hinzugefügt und daraus entfernt werden und $\cup, \cap, \setminus$ berechnet werden können.

Sorten (Wertebereiche): $\mathbb{N}$, $2^{\mathbb{N}}$, $\mathbb{B} = \{t, f\}$ (Wahrheitswerte)

Signatur: Funktionen zum Finden, Hinzufügen, Entfernen, ...

$$\Sigma_F = \left\{ \begin{array}{lll} \text{contains} : & 2^{\mathbb{N}} \times \mathbb{N} \rightarrow & \mathbb{B} \\ \text{add, remove} : & 2^{\mathbb{N}} \times \mathbb{N} \rightarrow & 2^{\mathbb{N}} \\ \text{isempty} : & 2^{\mathbb{N}} \rightarrow & \mathbb{B} \\ \emptyset : & & 2^{\mathbb{N}} \\ \text{union, cut, diff} : & 2^{\mathbb{N}} \times 2^{\mathbb{N}} \rightarrow & 2^{\mathbb{N}} \end{array} \right\}$$

Axiome (definieren die Semantik der Operationen)

$$\Phi = \left\{ \begin{array}{ll} \forall s \in 2^{\mathbb{N}} & (\text{isempty}(s) = t \leftrightarrow s = \emptyset), \\ \forall n \in \mathbb{N} & (\text{contains}(\emptyset, n) = f), \\ \forall s \in 2^{\mathbb{N}} \forall n \in \mathbb{N} & (\text{contains}(\text{add}(s, n), n) = t), \\ \forall s \in 2^{\mathbb{N}} \forall n \in \mathbb{N} & (\text{contains}(\text{remove}(s, n), n) = f), \\ \forall s \in 2^{\mathbb{N}} \forall m, n \in \mathbb{N} & (\text{add}(\text{add}(s, n), m) = \text{add}(\text{add}(s, m), n)), \\ \forall s \in 2^{\mathbb{N}} \forall n \in \mathbb{N} & (\text{add}(\text{add}(s, n), n) = \text{add}(s, n)), \dots \end{array} \right\}$$

Konkrete Datentypen (Implementierung) – Beispiele

verschiedene Möglichkeiten zur Repräsentation der Sorten:

- $\mathbb{B}$ Wahrheitswerte z.B. als $\{0, 1\}$ oder $\{-1, 1\}$
- $\mathbb{N}$ natürliche Zahlen in verschiedenen Zahlendarstellungen, z.B.
dezimal, binär, zu anderer Basis,
Maschinenzahlen (eingeschränkter Bereich)
- $2^{\mathbb{N}}$ (endliche) Mengen natürlicher Zahlen z.B. als
 - ▶ (sortierte) Folgen von Zahlen (mit / ohne Wiederholungen)
 - ▶ charakteristische Funktion $\chi_s : \mathbb{N} \rightarrow \{0, 1\}$ der Menge, d.h.
Zuordnung $\chi_s : \mathbb{N} \rightarrow \{0, 1\}$
 - ▶ $\{0, 1\}$ -Folgen variabler Länge (charakteristischer Vektor)
relevanter Teil der charakteristischen Funktion

abstrakter Datentyp: Signatur Σ und Axiome Φ ,

exakte Formulierung der Aufgabe, Spezifikation

konkreter Datentyp: Σ -Struktur, welche Φ erfüllt (Modell für Φ)

Umsetzung, Implementierung

Was bisher geschah

Modellierung von

Daten durch Mengen

Beziehungen (Zusammenhänge und Eigenschaften) durch Relationen,
Graphen und Funktionen

Anforderungen durch Logiken

Modellierung zusammenhängender Datenbereiche durch

algebraische Strukturen (konkrete Datentypen)
Träermengen, Funktionen (, Relationen)
auch mehrsortig

Modellierung von Eigenschaften (Anforderungen) in

Logiken oft Fragmente oder Erweiterungen von $\text{FOL}(\Sigma, \mathbb{X})$

Abstrakte Datentypen (Σ, Φ)
Syntax (Signatur Σ) und Semantik (Axiome Φ)
von (Software-)Schnittstellen

Konkrete Datentypen Σ -Strukturen in $\text{Mod}(\Phi)$ (Implementierungen)

Beispiele: ADT Himmelsrichtungen, Menge, konkrete Datentypen dazu

Modellierungsbeispiel Papierstapel

- ▶ Individuenmengen:
 - ▶ Blätter (oben einseitig bedruckt)
 - ▶ Stapel (von Blättern)
- ▶ Eigenschaft: (Stapel) „ist leer“
- ▶ Funktionen:
 - $\text{top} : \text{Stapel} \rightarrow \text{Blatt}$
(oberes Blatt auf dem Stapel)
 - $\text{pop} : \text{Stapel} \rightarrow \text{Stapel}$
(oberes Blatt vom Stapel entfernen)
 - $\text{push} : \text{Blatt} \times \text{Stapel} \rightarrow \text{Stapel}$
(Blatt oben auf den Stapel legen)
 - $\text{new} : \text{Stapel}$
(neuer leerer Stapel)

Eigenschaften der Stapel-Operationen

Operationen:

$\text{top} : \text{Stapel} \rightarrow \text{Blatt}$

$\text{pop} : \text{Stapel} \rightarrow \text{Stapel}$

$\text{push} : \text{Blatt} \times \text{Stapel} \rightarrow \text{Stapel}$

einige Zusammenhänge zwischen den Stapel-Operationen:

für alle Blätter (Stapelelemente) e und alle Stapel s gilt:

- ▶ Wird zuerst ein Blatt e auf einen Stapel s gelegt und dann von diesem Stapel das obere Blatt weggenommen, enthält man den ursprünglichen Stapel s
 $\text{pop}(\text{push}(e, s)) = s$
- ▶ $\text{top}(\text{push}(e, s)) = e$
- ▶ $\text{push}(\text{top}(s), \text{pop}(s)) = s$

Beobachtung:

Dieselben Eigenschaften gelten auch für Karten, Bücher-, Teller- und andere Stapel,

sind also **unabhängig vom Typ der Stapelelemente** (polymorph)

ADT Stack (Stapel, Keller)

Abstraktion vom Elementtyp (polymorph) und von der Realisierung (Trägersmengen, Implementierung der Operationen)

Sorten E : Elementtyp,
 $S(E)$: Stapel von Elementen vom Typ E ,
 $\mathbb{B} = \{f, t\}$ (Wahrheitswerte)

Signatur Σ :

top	: $S(E)$	$\rightarrow E$
pop	: $S(E)$	$\rightarrow S(E)$
push	: $E \times S(E)$	$\rightarrow S(E)$
new	:	$S(E)$
isEmpty	: $S(E)$	$\rightarrow \mathbb{B}$

Axiome , z.B.

$$\begin{aligned} \text{isEmpty}(\text{new}) &= t \\ \forall s \in S(E) \ \forall e \in E \quad (\text{isEmpty}(\text{push}(e, s)) &= f) \\ \forall s \in S(E) \ \forall e \in E \quad (\text{top}(\text{push}(e, s)) &= e) \\ \forall s \in S(E) \ \forall e \in E \quad (\text{pop}(\text{push}(e, s)) &= s) \\ \forall s \in S(E) \quad (\text{push}(\text{top}(s), \text{pop}(s)) &= s) \end{aligned}$$

konkrete Datentypen (Implementierungen) dazu in
LV Algorithmen und Datenstrukturen (2. Semester)

ADT Stack (Stapel, Keller)

Aus den Axiomen

$$\begin{aligned} \text{isEmpty}(\text{new}) &\stackrel{(1)}{=} t \\ \forall s \in S(E) \ \forall e \in E \quad (\text{isEmpty}(\text{push}(e, s)) &\stackrel{(2)}{=} f) \\ \forall s \in S(E) \ \forall e \in E \quad (\text{top}(\text{push}(e, s)) &\stackrel{(3)}{=} e) \\ \forall s \in S(E) \ \forall e \in E \quad (\text{pop}(\text{push}(e, s)) &\stackrel{(4)}{=} s) \\ \forall s \in S(E) \quad (\text{push}(\text{top}(s), \text{pop}(s)) &\stackrel{(5)}{=} s) \end{aligned}$$

lassen sich durch syntaktische Umformungen

(ohne Wissen über Realisierung, konkreten Datentyp)

weitere Eigenschaften ableiten, welche in jeder Realisierung gelten,

z.B. gilt $\text{top}(\text{pop}(\text{push}(a, \text{push}(b, s)))) = \text{top}(\text{push}(b, \text{new}))$ wegen

$$\begin{aligned} \text{top}(\text{pop}(\text{push}(a, \text{push}(b, s)))) &\stackrel{(4)}{=} \text{top}(\text{push}(b, s)) \stackrel{(3)}{=} b \\ &\stackrel{(3)}{=} \text{top}(\text{push}(b, \text{new})) \end{aligned}$$

$$\text{empty}(\text{new}) = t$$

Algebraische Datentypen – Kombination von Strukturen

einfache (einsortige) Datentypen, z.B. `int`, `bool`, `float`, ...

zusammengesetzte Datentypen

Konstruktion durch Operationen auf den Trägermengen:

Vereinigung $\cup$ von Datentypen

Beispiel Skat: $W = Z \cup B$

Produkt (kartesisches) $\times$ von Datentypen:

Beispiel Skat: $K = F \times W$

mehrfaches Produkt desselben Typs,

z.B. Tupel

Beispiel Skat:

Blatt in der Hand nach dem Geben $\in K^{10}$

Rekursion : rekursive Datentypen, z.B. Listen, Bäume

Funktion : zwischen Datentypen (Abbildung)

z.B. Multimengen $\mathbb{N}^U$, Maps B^A

Zusammengesetzte Datentypen: mehrsortige Strukturen

Spezifikation und Verifikation mit ADT

Jeder ADT **spezifiziert** eine Menge konkreter Datentypen:
genau alle Implementierungen (Σ -Strukturen), die alle Axiome des ADT erfüllen

Menge aller **korrekten** Implementierungen des ADT
= Modellmenge der Menge Φ der Axiome des ADT

Verifikation:

formaler Nachweis der Korrektheit von Implementierungen bzgl.
der Spezifikation (ADT)

oft durch Ableitungen in geeigneten Kalkülen
maschinelle Unterstützung durch Werkzeuge wie
z.B. Coq, pvs, Isabelle, Maude

(mehr dazu in Modulen im Master-Studium)

WH: Lösung von Aufgaben

Analyse der (informalen) Aufgabe

Modellierung Übertragung aller relevanten Informationen von der Realität in einen Modellbereich (geeignet gewählt) durch Analyse und Formalisierung der Eigenschaften von

Aufgabenbereich (Kontext): (strukturierte) Daten, Eigenschaften, Zusammenhänge

Eingabedaten: Typ, mögliche Werte, Einschränkungen

Ausgabedaten (Lösung): Typ und Anforderungen:
Einschränkungen
Zusammenhänge mit den Eingabedaten

Formalisierung in abstrakten Datentypen (ADT)

Lösung der Aufgabe im Modellbereich mit

vorhandenen Methoden, z.B.

SAT-Solver, SW-Bibliotheken

speziellen (eigens entwickelten) **Algorithmen**

(Realisierung in konkreten Datentypen, Implementierung)

Übertragung der Lösung aus Modellbereich in Realität

Algorithmen

(Wiederholung aus LV zu Programmierung)

Algorithmus: in Schritte geordnete Arbeitsvorschrift

- ▶ in einer **formalen Beschreibungssprache**
- ▶ **endlich** beschriebene
- ▶ **schrittweise** ausgeführte

Arbeitsvorschrift

zur Lösung einer (Berechnungs-)Aufgabe,
d.h. zur **Transformation** einer **Eingabe** in eine **Ausgabe**

zur **Ausführung** eines Algorithmus ist nötig:
Akteur (z.B. Maschine), welche die Beschreibungssprache
interpretieren kann

Beispiel: Summe der ersten n natürlichen Zahlen

informale Aufgabenstellung:

Addiere alle natürlichen Zahlen bis n .

Spezifikation (formale Aufgabenbeschreibung, Anforderungen):

Vorbedingung: Eingabe $n \in \mathbb{N}$

Nachbedingung: Ausgabe $s \in \mathbb{N}$ mit $s = \sum_{i=0}^n i$

verschiedene Algorithmen, welche diese Spezifikation erfüllen:

Algorithmus : Summe1

Eingabe : $n \in \mathbb{N}$

Ausgabe : $s \in \mathbb{N}$

$s \leftarrow 0$

für jedes $i \leftarrow 1, \dots, n :$

$s \leftarrow s + i$

Ende

Algorithmus : Summe2

Eingabe : $n \in \mathbb{N}$

Ausgabe : $s \in \mathbb{N}$

$s \leftarrow \frac{n(n+1)}{2}$

Struktur von Algorithmen

Konstruktion komplexer Berechnungsvorschriften aus

Grundbausteinen: elementare Algorithmen (Schritte), z.B.

- ▶ Zuweisung,
- ▶ Aufruf eines Unterprogrammes,
- ▶ Ein- oder Ausgabe (Interaktion)

Verknüpfungen von Algorithmen durch

sequentielle Ausführung (nacheinander)

parallele Ausführung (gleichzeitig, benötigt
mehrere Ausführende, z.B. Prozessoren)

bedingte Ausführung, Alternative (Verzweigung)

wiederholte Ausführung (Schleifen)

rekursive Ausführung

Blöcke , Unterprogramme

(Notation in Struktogramm, Pseudocode, ...)

Algorithmen-Entwicklung

1. Analyse der informalen Aufgabenstellung
2. (formale) Spezifikation:
Was (welche Berechnungsaufgabe) soll gelöst werden?
exakte (formale) Beschreibung der Aufgabe:
 - ▶ Anforderungen an Eingaben des Algorithmus
 - ▶ Anforderungen an Ausgaben des Algorithmus
 - ▶ Zusammenhang zwischen Ein- und Ausgabe
3. Entwurf des Algorithmus:
Wie soll es gelöst werden?
 - ▶ formale Darstellung der Arbeitsschritte
 - ▶ zu jedem Schritt:
 - ▶ Was wird getan? (Aktionen, Anweisungen, Verknüpfungen)
 - ▶ Womit wird es getan? (Daten)
 - ▶ Wie geht es weiter? (nächster Schritt)
4. Verifikation:
Nachweis der **Korrektheit** des Algorithmenentwurfes bzgl. der Spezifikation
5. Realisierung (Implementierung)

Algorithmen – Analyse der Aufgabe

Was soll gelöst werden?

- ▶ ist zu Beginn des Software-Entwicklungsprozesses oft noch nicht klar,
- ▶ zunächst grober Ansatz,
- ▶ wird schrittweise verfeinert,
- ▶ formale Darstellung fördert
 - ▶ Problemverständnis,
 - ▶ Abstraktion (Auswahl relevanter Eigenschaften),
 - ▶ Dokumentation während des Entwicklungsprozesses,
 - ▶ Ideen für Lösungsansätze

Algorithmen – Spezifikation

Ausgangspunkt: umgangssprachlich formulierte und oft ungenaue Aufgabenbeschreibung

Ergebnis: exakte und vollständige Definition des Problems

Spezifikation einer Berechnungsaufgabe:

korrekte formale Beschreibung des Zusammenhanges zwischen Eingaben und Ausgaben

Spezifikation einer Berechnungsaufgabe enthält

Vorbedingung: Anforderungen an die Eingaben

Nachbedingung: Anforderungen an die Ausgaben

Beispiel: Maximum-Suche

informale Aufgabenstellung:

Entwurf eines Verfahrens, welches in jeder Folge natürlicher Zahlen das Maximum findet

formale Spezifikation:

Vorbedingung: Eingabe $(x_1, x_2, \dots, x_n) \in \mathbb{N}^*$

Nachbedingung: Ausgabe $m \in \mathbb{N}$ mit

▶ $\forall i \in \{1, \dots, n\} : x_i \leq m$ und

▶ $\exists i \in \{1, \dots, n\} : m = x_i$

aus Spezifikation lässt sich folgender Algorithmus „ablesen“:

Algorithmus : Maximum

Eingabe : $x = (x_1, \dots, x_n) \in \mathbb{N}^*$

Ausgabe : $m \in \mathbb{N}$

$m \leftarrow 0$

für jedes $i \leftarrow 1, \dots, n$:

wenn $x_i > m$ **dann** $m \leftarrow x_i$

Ende

(effizientere Verfahren im Modul Algorithmen und Datenstrukturen)

Beispiel: größter gemeinsamer Teiler

Aufgabe: Zu zwei natürlichen Zahlen soll ihr größter gemeinsamer Teiler (ggT) berechnet werden.

Kontextwissen (Definitionen):

- ▶ t ist Teiler von x :

$$\forall (t, x) \in \mathbb{N}^2 : (t \mid x \leftrightarrow \exists k \in \mathbb{N} : (t \cdot k = x))$$

- ▶ Menge aller gemeinsamen Teiler von $x \in \mathbb{N}$ und $y \in \mathbb{N}$:

$$\forall (t, x) \in \mathbb{N}^2 : T(x, y) = \{t \in \mathbb{N} \mid (t \mid x) \wedge (t \mid y)\}$$

- ▶ größter gemeinsamer Teiler von $x \in \mathbb{N}$ und $y \in \mathbb{N}$

$$\forall (t, x, y) \in \mathbb{N}^3 : ((\text{ggT}(x, y) = t) \leftrightarrow ((t \in T(x, y)) \wedge (\forall s \in T(x, y)(s \mid t))))$$

Spezifikation (Anforderungen):

Vorbedingungen (an Eingaben): $x \in \mathbb{N}, y \in \mathbb{N}$

Nachbedingungen (an Ausgaben): $z \in \mathbb{N}$ mit $z = \text{ggT}(x, y)$

Beispiel: Algorithmus für ggT

Kontextwissen: (bekannte) Eigenschaften des ggT:

$$\begin{aligned}\forall x \in \mathbb{N} : & \quad \text{ggT}(x, x) = x \\ \forall (x, y) \in \mathbb{N}^2 : & \quad \text{ggT}(x, y) = \text{ggT}(y, x) \\ \forall (x, y) \in \mathbb{N}^2 : & \quad ((x > y) \rightarrow (\text{ggT}(x, y) = \text{ggT}(x - y, y)))\end{aligned}$$

führen zur Idee des (einfachen) Euklidischen Algorithmus:

Algorithmus : Größter gemeinsamer Teiler

Eingabe : $x \in \mathbb{N}, y \in \mathbb{N}$

Ausgabe : $\text{ggT}(x, y)$

solange $x \neq y$:

wenn $x > y$ **dann**

$x \leftarrow x - y$

sonst

$y \leftarrow y - x$

Ende

Ende

Rückgabe x

Beispiel: Sortieren

informale Aufgabenstellung:

Gesucht ist ein Verfahren, welches jede (endliche) Folge sortiert

Kontext (wird mitunter erst auf Nachfrage klar):

- ▶ Trägermenge (Elemente der Folge) (z.B. $\mathbb{N}$)
- ▶ totale Ordnung auf der Trägermenge, bzgl. welcher sortiert werden soll, (z.B. $\leq$ auf $\mathbb{N}$)
- ▶ auf- oder absteigend sortiert? (z.B. aufsteigend)

formale Spezifikation dieser Sortier-Aufgabe:

Vorbedingung: Eingabe $(x_1, x_2, \dots, x_n) \in \mathbb{N}^*$

Nachbedingung: Ausgabe $(y_1, y_2, \dots, y_n) \in \mathbb{N}^*$ mit

1. $y_1 \leq y_2 \leq \dots \leq y_{n+1}$ (aufsteigend geordnet) und
2. $(y_1, y_2, \dots, y_n)$ ist Permutation (Umordnung) der Eingabe $(x_1, x_2, \dots, x_n)$,

(verschiedene Verfahren dafür im Modul Algorithmen und DS)

Korrektheitsnachweise für Algorithmen

Beispiel Hoare-Kalkül (Tony Hoare 1969, Turing-Award 1980)
zur Spezifikation imperativer Programme / Algorithmen P
durch Hoare-Tripel $\{\varphi\}P\{\psi\}$ mit

Vorbedingung $\varphi \in \text{FOL}(\Sigma, X)$

Nachbedingung $\psi \in \text{FOL}(\Sigma, X)$

„... there are two ways of constructing a software design: One way is to make it so simple that there are obviously no deficiencies and the other way is to make it so complicated that there are no obvious deficiencies. The first method is far more difficult.“

Tony Hoare, 1980 ACM Turing Award Lecture

Ist der entworfene Algorithmus **korrekt**?

Erfüllt der entworfene Algorithmus die Spezifikation?

Verifikation: Nachweis, dass Entwurf die Spezifikation erfüllt
(Tests können das i.A. nicht)

Es gibt Verfahren / Werkzeuge, die Verifikation automatisch ausführen oder unterstützen.

(mehr dazu in LV Verifikation im Master-Studium)

Korrektheitsnachweise im Entwicklungsprozess

Anforderungen analysieren

Spezifikation (formale): **Vor- und Nachbedingungen**
(Voraussetzung für Korrektheitsnachweise und Tests)

Entwurf eines Algorithmus (Lösungsverfahren)
zur Abbildung von Eingabe- auf Ausgabedaten
als Struktogramm, Pseudocode, ...

Korrektheitsnachweis (Verifikation) des **Entwurfs**
(Algorithmus):

Nachweis, dass der Entwurf die Spezifikation
(Vorbedingungen, Nachbedingungen) erfüllt

Realisierung (Implementierung) des Algorithmus
Korrektheitsnachweis (Verifikation) der **Implementierung**:
formaler Nachweis, dass die Implementierung (Programm)
die Spezifikation für **jede** korrekte Eingabe (Vorbedingung)
erfüllt

Ausführung der Implementierung des Algorithmus (Programmes)

Tests: Überprüfung, ob die Implementierung (Programm) die
Spezifikation für **einige** spezielle Eingaben erfüllt

Was bisher geschah

Modellierung von

Daten durch Mengen (einfach, zusammengesetzt)

Beziehungen (Zusammenhänge und Eigenschaften)
durch Relationen, Graphen und Funktionen

Anforderungen (Spezifikation) an Daten, Eigenschaften,
Zusammenhänge, Algorithmen, ...
durch Logiken

Abstrakter Datentyp A : definiert Anforderungen

syntaktisch : Signatur Σ_A

semantisch : Axiome $\Phi_A \subseteq \text{FOL}(\Sigma_A, \mathbb{X})$

repräsentiert Menge konkreter Datentypen
(SW-Schnittstelle)

Konkreter Datentyp zu A : Σ -Struktur $\mathcal{S} \in \text{Mod}(\Phi_A)$
(Implementierung)

Spezifikation von Algorithmen

Vorbedingungen: Typ, Anforderungen an die Eingabe

Nachbedingungen: Typ, Anforderungen an die Ausgabe,
Zusammenhang mit der Eingabe

Modellierung von Abläufen

Abläufe sind charakterisiert durch

- ▶ Zustände, Daten
- ▶ Aktionen
- ▶ Übergänge zwischen Zuständen
abhängig von Aktion, aktuellem Zustand und Daten
- ▶ Startkonfiguration
- ▶ evtl. Endkonfigurationen, evtl. mit Ausgabe

Beispiele:

- ▶ Münzspiel (Ein- oder Zwei-Personen-Varianten)
 - ▶ Zustände $w \in \mathbb{N}^*$
 - ▶ Daten, z.B. Anzahl n der Münzen zu Beginn
 - ▶ Aktionen: Spielzüge entsprechend Spielregeln
 - ▶ Übergänge: Wirkung der Spielzüge
 - ▶ Startkonfiguration z.B. $0n0$
 - ▶ Endkonfigurationen z.B. $w' \in \{0,1\}^*$, Ausgabe: ... gewinnt
- ▶ Ausführung eines imperativen Programmes
 - ▶ Zustand: Speicherbelegung
 - ▶ Daten: Eingaben (zu Beginn oder laufend)
 - ▶ Aktionen: Anweisungen, ...
 - ▶ Übergänge: Änderung der Speicherbelegung bei Ausführung

Beispiel: Himmelsrichtungen

(Blickrichtung in) Himmelsrichtungen

Norden, Osten, Süden, Westen

mit Operationen für Rechts- und Linksabbiegen und Umlenken

Himmelsrichtungen (Zustände) $\{N, O, S, W\}$

Norden, Osten, Süden, Westen

Operationen (Aktionen) $\{R, L, U\}$

rechts drehen, links drehen, umdrehen

Ausführung einer **Aktion** bewirkt

Übergang (Relation) zwischen zwei **Zuständen**

d.h. jede Aktion $a \in A$ definiert eine zweistellige Relation $\delta(a)$ auf der Menge der Zustände,

im Beispiel:

$$\delta(R) \subseteq \{N, O, S, W\}^2$$

$$\delta(R) = \{(N, O), (O, S), (S, W), (W, N)\}$$

$$\delta(L) \subseteq \{N, O, S, W\}^2$$

$$\delta(L) = \{(N, W), (O, N), (S, O), (W, S)\}$$

$$\delta(U) \subseteq \{N, O, S, W\}^2$$

$$\delta(U) = \{(N, S), (O, W), (S, N), (W, O)\}$$

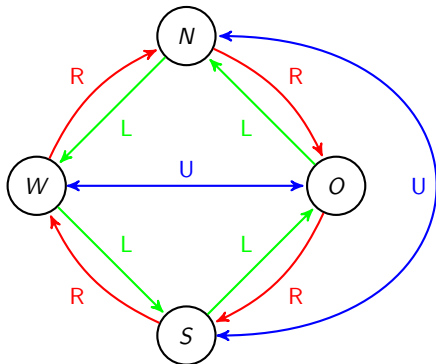
Übergänge zwischen Himmelsrichtungen

gemeinsame Darstellung dieser Relationen als Graph (V, E) mit Kantenfärbung

$V = \{N, O, S, W\}$ Menge der Zustände

$E = \delta(R) \cup \delta(L) \cup \delta(U)$ Kantenrelation

$A = \{R, L, U\}$ Menge der Aktionen als Kantenfarben



Zustandsübergangssysteme

Zustandsübergangssystem (Q, X, δ)

Q (endliche) Menge von **Zuständen**

X endliche Menge von **Aktionen**

$\delta : X \rightarrow (Q \times Q)$ **Übergangsrelationen**

δ ordnet jeder Aktion $a \in X$ eine Relation

$\delta(a) \subseteq Q \times Q$ zu

(Graph mit Kantenfärbung, Farben: Aktionen)

Spezifikation (formale Beschreibung von Anforderungen) von Zustandsübergangssystemen

Darstellung der Eigenschaften von

- ▶ Zuständen
- ▶ (zulässigen) Aktionen
- ▶ Wirkung der Aktionen (Zustandsübergänge)

als (prädikatenlogische) Formelmengen.

Modellierung mit Zustandsübergangssystemen

Anwendungsbeispiele:

- ▶ reale Automaten,
z.B. Getränke-, Fahrschein, Bankautomaten
- ▶ Verkehrssysteme,
z.B. Stellwerk, Ampelschaltungen
- ▶ Steuerung von Industrieanlagen
- ▶ Digitaltechnik, Schaltwerke
- ▶ Berechnungen,
z.B. durch Ausführung von Programmen
- ▶ Bedien-Oberflächen,
z.B. Folgen von Bedienoperationen
- ▶ Zeit- und Ablaufplanung
- ▶ Geschäftsprozesse
- ▶ Spiel-Abläufe

Beispiel: Münzschließfach

Aktionen: **A** aufschließen

Z zuschließen

O Tür öffnen

S Tür schließen

G Geld einwerfen

Eigenschaften: **g** bezahlt

o Tür offen

b belegt

Anforderungen (Spezifikation), z.B.

Am Münzschließfach sollen nur Abläufe (Wort $w \in \{A, Z, O, S, G\}^*$) möglich sein, die folgende Anforderungen (Eigenschaften) erfüllen:

- ▶ Offene Fächer können nie zugeschlossen werden.

$$\forall x(o(x) \rightarrow \neg \exists y Z(x, y))$$

- ▶ Für belegte Fächer kann nie Geld eingeworfen werden.

$$\forall x(b(x) \rightarrow \neg \exists y G(x, y))$$

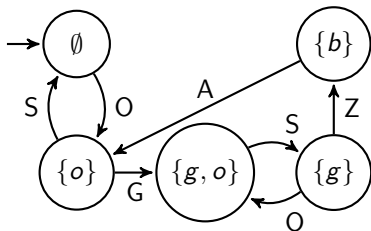
- ▶ Nicht bezahlte Fächer können nie zugeschlossen werden.

$$\neg \exists x(\neg g(x) \wedge \exists y Z(x, y))$$

Dazu werden auch häufig nichtklassische Logiken verwendet, z.B. Temporallogiken: (entscheidbare) FOL-Fragmente (mehr dazu in späteren LV zu Verifikation)

Beispiel Münzschließfach

Entwurf des Systems:



Beispiele für mögliche Abläufe (während eines Tages)

$w \in \{A, Z, O, S, G\}^*$:

- ▶ *OGSZAGSZA*: zwei vollständige Belegungszyklen
- ▶ *OSOSOS*: Tür wiederholt öffnen und schließen
- ▶ ε
- ▶ *OSOGSOSZASOSOGSZASO*

Jeder mögliche Ablauf erfüllt die Spezifikation (Anforderungen):

$$\Phi = \left(\begin{array}{l} \forall x(o(x) \rightarrow \neg \exists y Z(x, y)), \forall x(b(x) \rightarrow \neg \exists y G(x, y)), \\ \neg \exists x(\neg g(x) \wedge \exists y Z(x, y)) \end{array} \right)$$

Menge aller möglichen Abläufe ist reguläre Sprache
(Alphabet: Aktionen)

mehr dazu in den LV zur Theoretischen Informatik (INB 4. Sem., INM)

Endliche Automaten – Definition

NFA (nondeterministic finite automaton) $A = (X, Q, \delta, I, F)$ mit

X endliches Alphabet,

Q endliche Menge von Zuständen,

δ Übergangsrelationen $\delta : X \rightarrow (Q \times Q)$,

$I \subseteq Q$ Startzustände,

$F \subseteq Q$ akzeptierende Zustände.

Beispiel:

$A = (X, Q, \delta, \{0, 3\}, \{2, 3, 4\})$ mit

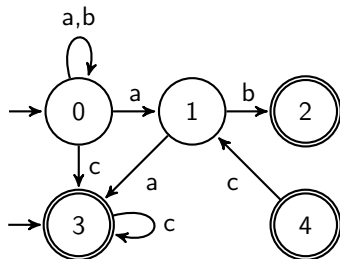
$$X = \{a, b, c\}$$

$$Q = \{0, 1, 2, 3, 4\}$$

$$\delta(a) = \{(0, 0), (0, 1), (1, 3)\}$$

$$\delta(b) = \{(0, 0), (1, 2)\}$$

$$\delta(c) = \{(0, 3), (3, 3), (4, 1)\}$$



Anwendung endlicher Automaten

Endliche Automaten können z.B.

- ▶ reguläre Sprachen akzeptieren,
d.h. bei Eingabe mit ja / nein antworten

Anwendungen z.B.

- ▶ Zulässigkeit von Email-Adressen
- ▶ Suchen von (auch mehreren) Zeichenketten in Texten
- ▶ Programmtransformation (z.B. im Compiler):
Syntax-Überprüfungen: Bezeichner, Zahl-Darstellungen,
korrekte Schlüsselwörter

Endliche Automaten können z.B. nicht

- ▶ Werte ausgeben
- ▶ Zeichenvorkommen zählen
- ▶ beliebig lange Zeichenketten speichern
- ▶ korrekte Klammerung erkennen
- ▶ rechnen, z.B. addieren

Automaten mit Ausgabe: Beispiel Kaffee-Automat

(einfacher) Kaffee-Automat:

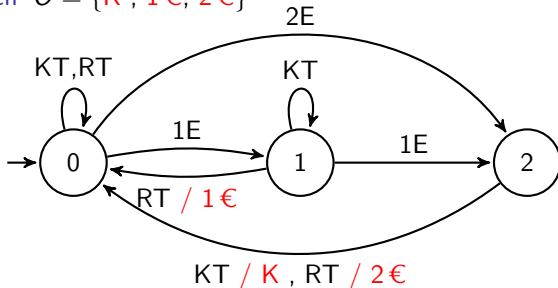
- ▶ liefert nach Einwurf von 2 € und Bedienung der Kaffee-Taste einen Becher Kaffee
- ▶ erlaubt Einwurf von Münzen zu 1 € und 2 €
- ▶ Rückgabe der gezahlten Münzen nach Bedienung der Rückgabetaste

formale Beschreibung als **Mealy-Automat** (Ausgabe bei Übergängen):

Zustände $Q = \{0, 1, 2\}$ (Anzahl der bisher gezahlten €)

Aktionen $A = \{1E, 2E, KT, RT\}$

Ausgaben $O = \{K, 1€, 2€\}$



Automaten mit Ausgabe: Beispiel Schieberegister

Pseudozufallszahlen (als Bitfolgen, z.B. für Verschlüsselung) lassen sich durch (linear rückgekoppelte) Schieberegister erzeugen

Beispiel: 3 Register mit Inhalten r_1, r_2, r_3 , in jedem Schritt:

Ausgabe r_3 und

Übergänge $r_1 r_2 r_3 \rightarrow (r_1 \text{ XOR } r_3) r_1 r_2$ (Rückkopplung)

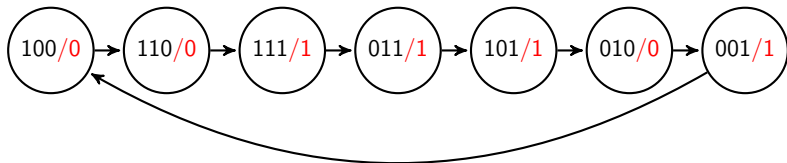
Eingabe Startwert $r_1 r_2 r_3 \in \{0, 1\}^3$ (seed)

formale Beschreibung als **Moore-Automat** (Ausgabe bei Zuständen):

Zustände $Q = \{001, 010, 011, 100, 101, 110, 111\}$

Aktionen $A = \{w\}$ (nur eine Aktion, wird nicht markiert)

Ausgaben $O = \{0, 1\}$



erzeugt mit seed 100 (pseudo-zufällige) Ausgabefolge: 00111010011...

Automaten mit zusätzlichem Speicher (Ausblick)

- ▶ Endliche Automaten sind zur Modellierung vieler Systeme / Abläufe nicht ausdrucksstark genug
- ▶ Ausdrucksstärke lässt sich durch zusätzlichen Speicher erhöhen
- ▶ Aktionen im Speicher (Lesen, Schreiben usw.) sind Nebenwirkungen der Zustandsübergänge

Typische Arten interner Speicher

- ▶ Zusatzinformation
z.B. Temperatursensor in Waschmaschine (nur Lesen)
Variablenbelegungen (Lesen und Schreiben)
- ▶ Warteschlange
Daten werden in der Reihenfolge verarbeitet, in der sie in den Speicher eingetragen wurden (first in, first out, FIFO)
z.B. Print-Queue
- ▶ Stack (Kellerautomaten)
Daten werden in der umgekehrten Eingangs-Reihenfolge verarbeitet (last in, first out, LIFO)
z.B. Socken und Schuhe an- und ausziehen, Palindrome, korrekte Klammerung
- ▶ unendliche Arbeitsbänder (Turing-Maschinen)
einfaches abstraktes Berechnungsmodell mit derselben Ausdrucksstärke wie aktuelle Rechentechnik

(mehr dazu in den LV zur Theoretischen Informatik)

Berechnungen als Zustandsübergangssysteme

imperative Programmierung (von-Neumann-Modell):

Programm (imperativ): Folge von Anweisungen

Ausführungsmodell: abstrakte Maschine

Zustand der abstrakten Maschine besteht aus

- ▶ Belegung der Variablen im Speicher und
- ▶ nächste Anweisung (Programmmähler)

Startzustand für Programm p und Eingabe i :

- ▶ Variablen im Speicher mit Werten aus i belegt,
- ▶ erste Anweisung von p (Programmmähler)

Berechnung (Ausführung) des Programmes p :

sequentielle Ausführung der Anweisungen in p

Modellierung durch (endliche oder unendliche) Folge von Zustandsübergängen (Rechenschritten)

Ausgabe: Speicherbelegung nach Ende der Berechnung

Semantik (Wirkung) eines imperativen Programmes p :

Abbildung (partielle Funktion) von Eingaben auf Ausgaben

und Nebenwirkungen (z.B. Änderung der Speicherbelegung)

Beispiel: Vertauschen mit Hilfsvariable

Aufgabe (informal): Vertauschen der Werte zweier Variablen

formale Spezifikation: **V:** Eingabe $a = A \in \mathbb{N}, b = B \in \mathbb{N}$

N: Ausgabe $a = B \in \mathbb{N}, b = A \in \mathbb{N}$

Zustände z.B. $(A, B, C) \in \mathbb{N}^3$ (unendlich viele), wobei
A Wert der Variable (Speicherplatz) a (B, C analog)

Übergänge definiert durch Anweisungen des Programmes

Bsp.: Programm Zustandsübergangssystem der Berechnung für Eingaben
 $a = 3, b = 5$ (Beispiel) $a = A, b = B$ (allgemein)

	$\rightarrow (3, 5, *)$	$\rightarrow (A, B, *)$
$c \leftarrow a$	$\downarrow (c \leftarrow a)$	$\downarrow (c \leftarrow a)$
	$(3, 5, 3)$	(A, B, A)
$a \leftarrow b$	$\downarrow (a \leftarrow b)$	$\downarrow (a \leftarrow b)$
	$(5, 5, 3)$	(B, B, A)
$b \leftarrow c$	$\downarrow (b \leftarrow c)$	$\downarrow (b \leftarrow c)$
	$(5, 3, 3)$	(B, A, A)

Verifikation des Programmes (Nachweis Korrektheit bzgl. Spezifikation)
durch Ausführung mit symbolischen Werten (manuell oder maschinell)
(mehr dazu in LV zu Verifikation)