Formale Methoden

Marcus Kracht
Fakultät LiLi
Universität Bielefeld
Postfach 10 01 31
D-33501 Bielefeld
marcus.kracht@uni-bielefeld.de

24. Januar 2018

Dieser Text ist die Grundlage der Vorlesungen Formale Methoden I und Formale Methoden II an der Universität Bielefeld. Kritik und Anregungen sind jederzeit willkommen.

Inhaltsverzeichnis

I Formale Methoden I: Mengen, Relationen, Sprachen 9

1 Mengen: Was sind sie und wie bekommt man sie? 11

2 Mengenoperationen 19

3 Relationen 29

4 Funktionen 39

5 Rekursion und Induktion 49

6 Wörter und Sprachen 61

7 Wie viele und welche Sprachen gibt es? 71

8 (Formale) Grammatiken 79

9 Reguläre Sprachen 91

10 Endliche Automaten 99

11 Reguläre Grammatiken 109

12 Kontextfreie Grammatiken und Satzstruktur 115

13 Struktur und Bedeutung 125

14 Anhang: Axiomatische Mengenlehre 131

II Formale Methoden 2: Logik 135

15 Einführung: Syntax und Semantik 137

16 Aussagen 141

17 Variable und Belegungen 151

18 Normalformen 161

19 Der Tableau-Kalkül 171

20 Folgerung 179

21 Ein Deduktionskalkül 187

22 Modallogik 199

23 Prädikatenlogik 205

24 Prädikatenlogik: Semantik I 217

25 Prädikatenlogik: Semantik II 225

26 Definierbarkeit 231

27 Substitution 241

28 Ein Kalkül 249

29 Syllogistik 253

Index 261

Literaturverzeichnis 265

Bevor es losgeht

Bevor ich mit dem eigentlichen Inhalt beginne, möchte ich einige Dinge zum Text und zu der Methodik sagen. Ein wesentliches Merkmal des Textes ist, dass er relativ „trocken” ist, das heißt zum Beispiel, dass ich jede gemachte Behauptung beweise, sofern sie nicht ein sogenanntes Axiom ist. Axiome sind in der mathematischen Praxis Setzungen, genauso wie Spielregeln. Wer sie nicht anerkennt, spielt nicht dasselbe Spiel. Natürlich sind diese Setzungen motiviert, und die Motivation erkläre ich auch. Jedoch kann man das Spiel auch dann spielen, wenn man die Motivation nicht versteht oder ablehnt. Der Gehalt der Sätze wird oft durch die Beweise erst wirklich klar. Denn der Beweis, wenn er denn einer ist, enthüllt, was es zu zeigen gibt. Ich erwarte allerdings von niemandem, dass sie diese Beweise auswendung kennen oder gar wiederholen können. Dennoch gibt es sicher einige, die wissen wollen, warum denn nun der behauptete Sachverhalt gilt. Deswegen gebe ich den Grund auch an, und der Grund besteht in der Regel in einem Beweis. Ich bitte deswegen um Nachsicht, denn dieser Text soll ja einem bunten Publikum dienen. Aus der Vorlesung wird ohnehin klar werden, worauf es ankommt.

Die hier verwendete Notation ist wie folgt. Ich unterscheide symbolisch zwischen zwei Symbolen für die Gleichheit: „=” und „:=”. Das erste steht für eine Gleichheit, die behauptet wird (etwa in „$3 + 4 = 7$”) also eine Aussage. Das zweite steht für eine Gleichheit durch Setzung, etwa in „$\langle M, N\rangle := \{M, \{M, N\}\}$”), also eigentlich eine Anweisung. Dies ist analog zu Computersprachen, bei denen „:=” dafür genommen wird, um einer Variablen einen Wert zuzuweisen („$x := \sqrt{7}$”), während „=” bei Abfragen genommen wird (etwa in „if x=y then”). Dies ist deswegen nützlich, weil nicht immer aus dem Text selber deutlich wird, ob es sich nun um eine Setzung handelt oder nicht.

Ein Begriff wird fett gedruckt, wenn er definiert wird. Üblicherweise wird ein Begriff auch dann zum ersten Mal verwendet, wenn er gerade definiert wird. Dies ist aber nicht immer der Fall, und solche Vorkommen werden besonders gekennzeichnet (zum Beispiel durch Schrägdruck).

Genauso wird eine Notation wird erst dann eingeführt bevor sie zum ersten Mal verwendet werden soll. Es sei hier betont, dass die genaue Notation oft unerheblich ist. So schreiben viele Autoren anstelle des von mir benutzten Differenzzeichens „–" den Schrägstrich „\"; sie schreiben also „$M\backslash N$" anstelle von „$M-N$", wie ich hier schreibe. Ebenso schreibe ich „$\{x : \cdots\}$", wo viele „$\{x \mid \cdots\}$" schreiben. Es bleibt Ihnen überlassen, ob Sie die von mir vorgeschlagene Notation übernehmen wollen oder eine andere. Ich gebe gelegentlich Alternativen an, und diese sind dann frei verfügbar. Sie können stets auch selber eine andere, nicht erwähnte Notation verwenden, *aber Sie müssen diese dann einwandfrei definieren. Eine alternative Notation, die nicht offiziell erwähnt wurde, kommt stets mit dem Preis, dass man sich erklären muss*. Das wäre so, als würde man Worte verwenden, deren Bedeutung die anderen nicht kennen.

Schließlich noch etwas zu dem formalen Überbau. Die Vorlesung behandelt die Themen Mengen, Sprachen, Grammatiken und Logik, Aussagen- wie Prädikatenlogik. Dabei werde ich unter anderem zeigen, wie man grundsätzlich alles als Menge verstehen kann. (Das ist das Projekt von Bourbaki, das dann als Neue Mathematik die Schulen erobert hat.) Dabei werde ich manchmal sagen: *soundso ist eine Menge folgender Gestalt*. Etwa so: *Das Paar* $\langle M, N\rangle$ *ist die Menge* $\{M, \{M, N\}\}$. Das ist keineswegs so zu verstehen, dass das Paar $\langle M, N\rangle$ ganz wörtlich die angegebene Menge ist. Wenn man das alles so streng nehmen würde, käme man auf keinen grünen Zweig. Sondern vielmehr gebe ich damit Folgendes zu verstehen: man kann im Prinzip auch ohne Paare auskommen; wenn wir anstelle des Paares $\langle M, N\rangle$ jetzt immer die Menge $\{M, \{M, N\}\}$ nehmen, bleibt eigentlich alles beim Alten. Und zwar insofern, als die neue Definition des Paares als Menge alles das erfüllt, was wir von dem Paar $\langle M, N\rangle$ erwarten. Sie sind funktional gleich. Man kann dies sehr hübsch mit Computern anschaulich machen. Ein Text ist eine Folge von Buchstaben. Diese Folge ist in einer Datei abgespeichert. Nun können wir sagen: ein Text ist ein Zeichen-Array (und so wird es auch in einigen Sprachen tatsächlich gehandhabt) aber oft nur auf der oberen Ebene. Wie es wirklich im Computer aussieht, wenn die Datei mal abgelegt wird oder das Programm in Maschinensprache übersetzt ist, das ist sehr unterschiedlich. Aber wir müssen darüber nicht wirklich Bescheid wissen, weil das Verhalten, das wir erwarten, stets das Gleiche ist. Und so bedienen wir uns einfach einer gewissen „Realisierung", welche uns das Gewünschte liefert, und erlauben einfach, dass die „wirklichen" Gegenstände durchaus anders sind.

Die Sprache, die ich verwende, ist zwar die deutsche Umgangssprache. Dennoch gibt es ein paar Eigenheiten, die ich vorab erklären muss. Zunächst einmal wird die logische Struktur von Behauptungen sehr ernst genommen und unterliegt gewissen Beschränkungen, die stets vorausgesetzt werden.

Bedingungen Eine Behauptung der Form „Wenn A so B” gilt als richtig, wenn A falsch ist. Ebenso gilt sie als richtig, wenn B wahr ist. Alle von mir gemachten Behauptungen, besonders aber die in Definitionen und Sätzen, sind so zu verstehen.

Alternativen Eine Behauptung der Form „A oder B” gilt als richtig, wenn A wahr ist, egal, ob dann auch B wahr ist oder nicht. Sie gilt ebenso als richtig, wenn B wahr ist, egal, ob dann A auch wahr ist oder nicht. *Disjunktionen sind immer inklusiv.*

Variable Verschiedene Variable müssen nicht verschiedene Werte haben; ja, es ist sogar so, dass der Fall, dass ihre Werte gleich sind, oft sogar besonders hervorgehoben ist. Heißt es zum Beispiel „Seien M und N Mengen.”, so ist nicht notwendig M verschieden von N. Es ist ausdrücklich erlaubt, dass M und N dieselbe Menge sind. Zwar ist dann der Plural nicht ganz angemessen, aber dies wird hier ignoriert beziehungsweise als rein formale morphologische Kongruenz angesehen, die nicht inhaltlich relevant ist.

Natürliche Zahlen Der Bereich der natürlichen Zahlen wird mit $\mathbb{N}$ bezeichnet. Es handelt sich hier um die Zahlen 0, 1, 2, 3, und so weiter. Wir fangen also mit 0 an zu zählen! Eine formale Definition wird noch erfolgen.

Ganze Zahlen Der Bereich der ganzen Zahlen wird mit $\mathbb{Z}$ bezeichnet. Er schließt die natürlichen Zahlen und die Zahlen $-n$, wo n eine ntürliche Zahl ist, ein.

Rationale Zahlen Der Bereich der rationalen Zahlen wird mit $\mathbb{Q}$ bezeichnet. Eine Zahl ist rational, wenn sie eine Darstellung m/n hat, wo m und n ganze Zahlen sind und $n \neq 0$.

Reelle Zahlen Der Bereich der reellen Zahlen wird mit $\mathbb{R}$ bezeichnet.

Der vorliegende Text ist mit Sorgfalt geschrieben worden. Entsprechend muss man ihn auch mit Sorgfalt lesen. Insbesondere die Definitionen sind oft das Ergebnis einer langen mathematischen Tradition, bei der jedes Detail abgewogen wurde. Die Formulierung wird nicht dem Zufall überlassen. Deshalb ist eine Exegese unbedingt notwendig, und ich versuche, eine solche auch mitzuliefern. Ich lege allerdings allen ans Herz, Definitionen nicht schnell zu lesen. Sie sind zentral, weil sie die Bedeutung der Begriffe und Notation festlegen. Wer sie nicht entsprechend würdigt, versteht nicht, was im Folgenden gesagt wird.

Teil I

Formale Methoden I: Mengen, Relationen, Sprachen

Kapitel 1

Mengen: Was sind sie und wie bekommt man sie?

Mengen sind die grundlegendsten Objekte in der Mathematik und zudem sehr einfach. Prinzipiell lässt sich die gesamte Mathematik mit Hilfe von Mengen darstellen, sodass die Mengenlehre innerhalb der Mathematik eine zentrale Grundlagendisziplin ist. Eine der wichtigsten Errungenschaften der Mengenlehre ist eine Aufklärung des Anzahl- und Ordnungsbegriffs. Wie wir noch später sehen werden, gibt es unter unendlichen Gesamtheiten durchaus Größenunterschiede. Unendlich ist also nicht gleich unendlich. Dies ist eine erstaunliche Sache, auf die man ohne ein gewisses Maß an Abstraktion sicher niemals kommen würde.

Eine Menge ist laut Georg Cantor wie folgt definiert.

> Unter einer „Menge“ verstehen wir jede Zusammenfassung M von bestimmten wohlunterschiedenen Objekten m unserer Anschauung oder unseres Denkens (welche die „Elemente“ von M genannt werden) zu einem Ganzen.

Cantor selbst benutzte statt des Begriffs Menge zunächst Wörter wie „Inbegriff“ oder „Mannigfaltigkeit“; von Mengen und Mengenlehre sprach er erst später.

Diese Definition bietet reichlich Stoff zum Nachdenken. Zunächst einmal weise ich darauf hin, dass sie nicht von realen Gegenständen spricht, sondern von Gegenständen unserer Anschauung oder unseres Denkens. Das hat Folgen, auf die ich unten zurückkommen werde. Und dann heißt es, diese Gegenstände sollen *wohlunterschieden* sein. Was das im Einzelnen bedeutet, sei dahingestellt. Man mache sich jedoch klar, dass zur Unterscheidung von Gegenständen zunächst einmal gehört, dass sie bereits da sind. Auch dies wiederum hat weitreichende Folgen (unter anderem diese, dass die Mengenlehre, die Cantor konzipiert hatte, keineswegs widersprüchlich ist; diese Ehre kommt den späteren Entwürfen zu, unter anderem dem von Frege).

Beginnen wir aber einmal von vorn. Nehmen wir mal an, wir haben folgende Gegenstände: $\heartsuit$, $\clubsuit$, $\spadesuit$ und $\diamondsuit$. Diese sind wohlunterschieden, und so können wir irgendwelche von ihnen zu einer Menge zusammenfassen. So können wir etwa $\heartsuit$ und $\spadesuit$ zusammenfassen. Die Menge, die wir so erhalten, wird mit $\{\heartsuit, \spadesuit\}$ bezeichnet. Die Klammern sind also dazu da, die Zusammenfassung zu symbolisieren. Innerhalb der Klammern werden die Gegenstände durch Kommata voneinander getrennt.

Bezeichen wir mal diese Menge mit M. Es ist also $M = \{\heartsuit, \spadesuit\}$. Dann ergeben sich uns folgende Fragen:

① Ist denn $M = \{\spadesuit, \heartsuit\}$, das heißt, kommt es bei der Nennung auf die Reihenfolge an?

② Ist denn $M = \{\heartsuit, \spadesuit, \heartsuit\}$, das heißt, kommt bei der Bezeichnung auf Mehrfachnennungen an?

③ Ist diese Menge jetzt gleich irgendeinem der vorher gegebenen Gegenstände oder verschieden von ihnen?

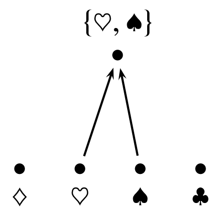
④ Was ist, wenn ich nur einen einzigen Gegenstand zu einer Menge zusammenfasse?

⑤ Kann ich auch gar keinen Gegenstand in meine Menge tun?

⑥ Wie viele Mengen kann ich denn auf diese Weise bekommen?

Ich werde der Reihe nach auf diese Fragen eingehen. Beginnen wir mit der ersten Frage, der Frage ①. Die Reihenfolge ist bei der Nennung der Elemente nicht erheblich. Dies lässt sich nicht so ohne Weiteres rechtfertigen; es ist einfach eine Definition, die Definition von Zusammenfassung. Worauf es nämlich bei der Menge lediglich ankommt, ist, ob ein gegebener Gegenstand in dieser Menge ist oder nicht. In der Menge M ist der Gegenstand $\heartsuit$ und der Gegenstand $\spadesuit$, und ebenso verhält es sich mit der Menge $\{\spadesuit, \heartsuit\}$. Dies beantwortet auch schon die zweite Frage: es kommt nicht darauf an, ob der Gegenstand mehrfach vorkommt. Also ist M auch gleich der Menge $\{\heartsuit, \spadesuit, \heartsuit\}$. Dabei mache man sich klar, dass nicht der Gegenstand selbst zweimal in $\{\heartsuit, \spadesuit, \heartsuit\}$ vorkommt; sondern er wird zweimal *genannt*. Die Zeichenfolge „$\{\heartsuit, \spadesuit, \heartsuit\}$" ist ja in Wirklichkeit nur ein Symbol, das eine

Abbildung 1.1: Mengen als Graphen I



Menge bezeichnen soll. Und es kann durchaus vorkommen, dass ein und derselbe Gegenstand durch mehrere Symbole gekennzeichnet wird (etwa „$1+2$” und „3”). Bei der Frage ③ ist es jetzt etwas schwieriger. Zunächst einmal deckt die Definition den Fall ab, wo meine Menge nur einen einzigen Gegenstand enthält. Sei etwa $N := \{\heartsuit\}$. Ist dann $N = \heartsuit$ (ohne Mengenklammern)? Wenn ja, dann müsste der Gegenstand $\heartsuit$ sich selbst enthalten. Das ist aber nicht der Fall. Die Mengen sind stets ein „Überbau”, sie können nicht identisch sein mit den Gegenständen, aus denen sie zusammengefasst werden.

Kann man denn nun auch *gar keinen Gegenstand* in einer Menge haben? Auch das geht. Diese Menge hat einen besonderen Namen: sie heisst die **leere Menge** oder **Nullmenge** und wird mit „$\varnothing$” bezeichnet (manche schreiben auch „$\{\}$”, aber das ist sehr selten).

Wie viele Mengen man so bekommt, ist schnell ausgerechnet. Wir haben vier Gegenstände, und eine Menge ist vollständig charakterisiert, wenn für jeden Gegenstand klar ist, ob er in der Menge ist oder nicht. Wir bekommen so $2^4 = 16$ Mengen, die ich in Tabelle 1.1 alle aufliste. Ich gebe hier noch eine (von vielen) Arten an, sich eine Menge vorzustellen. Die Menge $\{\heartsuit, \spadesuit\}$ wird symbolisiert durch einen Punkt, ebenso ihre Elemente, $\heartsuit$ und $\spadesuit$. Von den Elementen zeichnen wir einen Pfeil zu der Menge (siehe Abbildung 1.1). Eine solche Zeichnung heißt auch ein *Graph*. Ich habe noch die beiden anderen Gegenstände eingezeichnet, allerdings ohne Pfeil. Damit ist klar, dass sie in der Menge nicht enhalten sind. Die Menge ist durch ihre Elemente eindeutig bestimmt, deswegen kann man auf den Zusatz $\{\heartsuit, \spadesuit\}$ verzichten.

Jetzt mag man denken: das wär’s doch eigentlich, mehr geht nicht. Das ist aber ein Irrtum, denn jetzt geht es erst richtig los! Denn unsere Mengen sind ja jetzt auch Gegenstände unserer Anschauung, und die Mengen der Tabelle 1.1 sind erstens verschieden von unseren ursprünglichen Gegenständen, und zweitens sind

Tabelle 1.1: Alle sechzehn Mengen, die wir aus den Gegenständen $\spadesuit$, $\heartsuit$, $\clubsuit$, $\diamondsuit$ formen können

$\diamondsuit$	$\heartsuit$	$\spadesuit$	$\clubsuit$	Menge
$-$	$-$	$-$	$-$	$\varnothing$
$-$	$-$	$-$	$+$	$\{\clubsuit\}$
$-$	$-$	$+$	$-$	$\{\spadesuit\}$
$-$	$-$	$+$	$+$	$\{\spadesuit, \clubsuit\}$
$-$	$+$	$-$	$-$	$\{\heartsuit\}$
$-$	$+$	$-$	$+$	$\{\heartsuit, \clubsuit\}$
$-$	$+$	$+$	$-$	$\{\heartsuit, \spadesuit\}$
$-$	$+$	$+$	$+$	$\{\heartsuit, \spadesuit, \clubsuit\}$
$+$	$-$	$-$	$-$	$\{\diamondsuit\}$
$+$	$-$	$-$	$+$	$\{\diamondsuit, \clubsuit\}$
$+$	$-$	$+$	$-$	$\{\diamondsuit, \spadesuit\}$
$+$	$-$	$+$	$+$	$\{\diamondsuit, \spadesuit, \clubsuit\}$
$+$	$+$	$-$	$-$	$\{\diamondsuit, \heartsuit\}$
$+$	$+$	$-$	$+$	$\{\diamondsuit, \heartsuit, \clubsuit\}$
$+$	$+$	$+$	$-$	$\{\diamondsuit, \heartsuit, \spadesuit\}$
$+$	$+$	$+$	$+$	$\{\diamondsuit, \heartsuit, \spadesuit, \clubsuit\}$

sie auch noch untereinander verschieden. Denn es gilt Folgendes: zwei Mengen M und N sind genau dann gleich, wenn sie die gleichen Gegenstände enthalten; das bedeutet: Gegeben irgendein beliebiger Gegenstand m, so ist m genau dann in M, wenn er in N ist. Damit wir uns an die formale Schreibweise gewöhnen, werde ich dies in der nötigen Präzision aufschreiben.

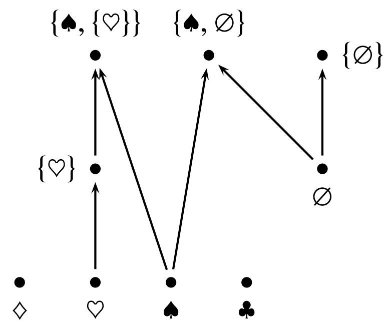
Definition 1.1 (Element) *Es sei M eine Menge. Wir schreiben „$m \in M$”, um zu sagen, dass m ein Element von M ist. Es sei N eine weitere Menge. Genau dann ist M gleich N, in Zeichen „$M = N$”, wenn für alle m gilt: $m \in M$ genau dann, wenn $m \in N$.*

(1.1) $$M = N \leftrightarrow (\forall m)(m \in M \leftrightarrow m \in N)$$

Hierbei ist $\leftrightarrow$ die Abkürzung von „genau dann, wenn” und $(\forall m)$ die Abkürzung von „für alle m”.

Indem wir nun erklären, dass auch Mengen Gegenstände unserer Anschauung sind, haben wir nunmehr unseren Objektbereich erweitert. Zu den Gegenständen

Abbildung 1.2: Mengen als Graphen II



zählen jetzt nicht allein die 4 vorher gegebenen Dinge—nennen wir sie die **ersten** Gegenstände—, sondern auch die daraus geformten 16 Mengen. Das macht jetzt 20 Gegenstände, denn die ersten Gegenstände sind verschieden von den Mengen, und natürlich auch untereinander. Und aus diesen 20 Gegenständen können wir nun wiederum in der gleichen Weise Mengen machen!

Dabei geht es jetzt in großen Schritten weiter. Aus insgesamt 20 Gegenständen kann man $2^{20} = 1048576$ Mengen basteln, zusätzlich zu den alten Mengen. (Eine einzige Ausnahme gibt es aber, vielleicht ahnen Sie schon, welche dieser Mengen nicht neu ist.) Neu hinzu kommen viele neue und zum Teil ganz ungewöhnliche Mengen wie $\{\spadesuit, \{\heartsuit\}\}$, $\{\{\diamondsuit\}\}$, $\{\spadesuit, \varnothing\}$, oder sogar $\{\varnothing\}$. Dabei ist $\{\varnothing\}$ *nicht* dasselbe wie $\varnothing$. Denn $\{\varnothing\}$ ist nicht leer; sie enthält ein (und nur ein) Element, die leere Menge. Dies ist sehr gewöhnungsbedürftig. Wir dürfen auf keinen Fall $\{M\}$ mit M gleichsetzen, was immer auch M sei. Deswegen ist jetzt $\{\{\diamondsuit\}\}$ weder gleich mit $\{\diamondsuit\}$ noch gleich mit $\diamondsuit$, sondern alle drei sind verschiedene Objekte. Ich habe zur Verdeutlichung einige dieser Mengen in einen Graphen eingezeichnet (Abbildung 1.2).

Die hier beschrieben Art der Mengenbildung war die von Cantor beschriebene. Jedoch ist diese Art zu umständlich. Nehmen wir mal an, wir wollen die Menge P aller Primzahlen bilden. Diese sähe rein intuitiv etwa so aus:

(1.2) $$P := \{2, 3, 5, 7, \cdots\}$$

Das Problem an dieser Definition sind die drei Punkte. Diese sagen nichts weiter wie: *und so weiter*. Aber wie genau geht es denn weiter? Das weiß man nicht wirklich. Es gibt nämlich ungeheuer viele Folgen von Zahlen, die so anfangen wie die oben genannte. Woher wissen wir denn jetzt, welche von ihnen gemeint ist? Um Fragen solcher Art auszuschließen, führen wir folgende Schreibweise ein.

Definition 1.2 (Komprehension) *Es sei $\varphi(x)$ eine Eigenschaft. Dann bezeichnet $\{x : \varphi(x)\}$ diejenige Menge, deren Gegenstände genau diejenigen sind, die φ erfüllen.*

Das kann man formal wie folgt aufschreiben:

(1.3) $\quad (\forall y)(y \in \{x : \varphi(x)\} \leftrightarrow \varphi(y))$

Man liest das in etwa so: „Für alle y gilt: genau dann ist y in der Menge aller x, die φ erfüllen, wenn y φ erfüllt.” Inhaltlich sollte das klar sein. Mit anderen Worten: Element der Menge $\{x : \varphi(x)\}$ zu sein heißt nichts Anderes, als φ zu erfüllen. Also haben wir jetzt die Möglichkeit, wie folgt zu schreiben.

(1.4) $\quad P := \{x : x \text{ ist Primzahl}\}$

Natürlich setzt dies voraus, dass der Begriff „ist Primzahl” eindeutig erklärt ist. Habe ich eine Definition des Begriffs (etwa den, dass die Zahl genau zwei Teiler hat, nämlich 1 und die Zahl selbst) so kann ich diese natürlich ebensogut benutzen.

(1.5) $\quad P = \{x : \ x \text{ hat genau 2 Teiler}\}$

Es kommt nicht darauf an, welche Beschreibung man wählt, vielmehr ist es nur wichtig, dass sie eindeutig ist. Auf diese Weise können wir jetzt jeden Begriff dazu benutzen, um Mengen zu definieren. Die Menge aller Autos ist dann zum Beispiel beschrieben durch $\{x : x \text{ ist Auto}\}$. Ich mache darauf aufmerksam, dass „x” eine sogenannte **Variable** ist. Das bedeutet, dass der Bedeutungsinhalt nicht feststeht, wie etwa auch bei $f(x) = x^2$ nicht gesagt wird, was x ist. Und so wie die Funktion $f(y) = y^2$ dieselbe Funktion ist wie $f(x) = x^2$, so gilt auch:

(1.6) $\quad \{x : x \text{ ist Auto}\} = \{y : y \text{ ist Auto}\}$

Der Name der Variable ist also unerheblich.

Für manche Mengen gibt es alternative Schreibweisen. Zum Beispiel ist $\{x : x \text{ ist eine gerade Primzahl}\}$ nichts anderes als die Menge $\{2\}$. Ferner ist

(1.7) $\quad \{x : x \text{ ist gerade und ungerade}\} = \varnothing$

Das hatten wir im Grunde genommen oben schon gesehen: dass eine Menge durchaus verschieden ist von den Symbolen, die wir verwenden, um sie zu benennen. Und die Namensgebung ist keineswegs systematisch und eindeutig, sondern eine einzelne Menge kann auf ganz viele verschiedene Weisen beschrieben werden.

Übungen

[1] Es sei der Grundbereich die Gesamtheit aller Primzahlen < 20. Geben Sie diesen an. Geben Sie zudem folgende Mengen aus dieser Gesamtheit konkret an, indem Sie ihre Elemente aufzählen.

1. $\{u : u$ hat eine gerade Quersumme$\}$
2. $\{v : v$ ist nicht durch 3 teilbar$\}$
3. $\{w : w$ ist größer als $10\}$
4. $\{x : x$ ist kein Teiler von $210\}$
5. $\{y : y$ hat eine gerade Quersumme$\}$

Sind diese Mengen alle verschieden?

[2] Können Sie eine (sprachliche) Beschreibung der Menge $\{3, 5, 7, 11, 13, 17, 19\}$ wie in [1] geben?

[3] Zeichnen Sie den Graphen der Menge $\{\{3, \{5\}\}, 5, \{2\}\}$.

[4] Sagen Sie für die folgenden vier Mengenpaare, ob die beiden Mengen gleich oder verschieden sind (und warum).

1. $\{\mathtt{y}, \mathtt{r}\}$ und $\{\mathtt{r}, \mathtt{y}\}$.
2. $\{\mathtt{y}, \mathtt{r}, \mathtt{u}\}$ und $\{\mathtt{r}, \mathtt{y}\}$.
3. $\{\mathtt{y}, \mathtt{y}\}$ und $\{\{\mathtt{u}\}, \mathtt{y}\}$.
4. $\{\emptyset, \mathtt{y}\}$ und $\{\mathtt{y}\}$.

Kapitel 2

Mengenoperationen

Aus Mengen können wir auf vielfältige Weise neue Mengen erschaffen. Dazu definieren wir spezielle Operationen, die genau dies bewerkstelligen sollen. Es seien dazu M und N Mengen; dann definieren wir den **Schnitt** $M \cap N$, die **Vereinigung** $M \cup N$ und das **relative Komplement von** N **in** M wie folgt.

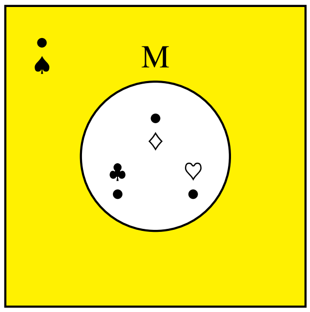
$$\begin{array}{rcl} M \cap N & := & \{x : x \in M \text{ und } x \in N\} \\ M \cup N & := & \{x : x \in M \text{ oder } x \in N\} \\ M - N & := & \{x : x \in M \text{ und nicht } x \in N\} \end{array} \tag{2.1}$$

Anstelle von “$M - N$” schreibt man auch oft “$M \backslash N$”. Ferner schreiben wir “$x \notin M$” anstelle von “nicht $x \in M$”. Dies erlaubt also, aus gegebenen Mengen neue zu machen. Zum Beispiel ist $\{\heartsuit, \diamondsuit\} \cap \{\spadesuit, \heartsuit\} = \{\heartsuit\}$, und $\{\heartsuit, \diamondsuit\} \cup \{\spadesuit, \heartsuit\} = \{\spadesuit, \heartsuit, \diamondsuit\}$; $\{\heartsuit, \diamondsuit\} - \{\spadesuit, \heartsuit\} = \{\diamondsuit\}$. Noch ein Beispiel. Ist M die Menge der Autos und N die Menge der roten Dinge, so ist $M \cap N$ die Menge der roten Autos. $M \cup N$ ist dann die Menge der Dinge, die rot sind oder ein Auto; und $M - N$ ist die Menge der nichtroten Autos. Ein drittes Beispiel. Ist M die Menge der Kinder und N die Menge der Mädchen, so ist $M \cap N$ die Menge der Mädchen (also N), $M \cup N$ ist die Menge der Kinder (also M) und $M - N$ die Menge der Jungen.

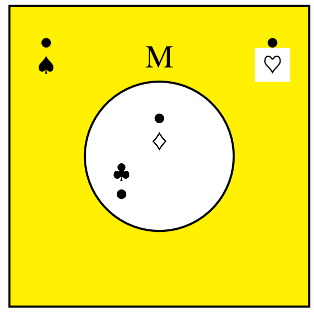
Ich zeige auch noch eine weitere Methode auf, Mengen zu veranschaulichen, die sogenannten **Euler-Venn Diagramme**. Dabei werden Mengen durch Kreise bezeichnet, Elemente in der Menge werden innerhalb des Kreises platziert, Elemente außerhalb der Menge außerhalb des Kreises. In (2.1)(a) zum Beispiel ist die Menge $\{\diamondsuit, \heartsuit, \clubsuit\}$ dargestellt. Dazu werden alle Elemente des Grundbereichs, die in der Menge sind, innerhalb des Kreises gestellt, alle anderen außerhalb des Kreises. Dies erlaubt, selbst die leere Menge darzustellen ((2.1)(c)).

Abbildung 2.1: Euler-Venn Diagramm

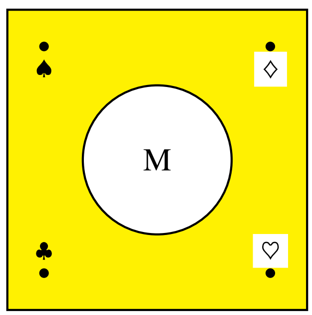
(a) Die Menge $\{\diamondsuit, \heartsuit, \clubsuit\}$



(b) Die Menge $\{\diamondsuit, \clubsuit\}$



(c) Die leere Menge



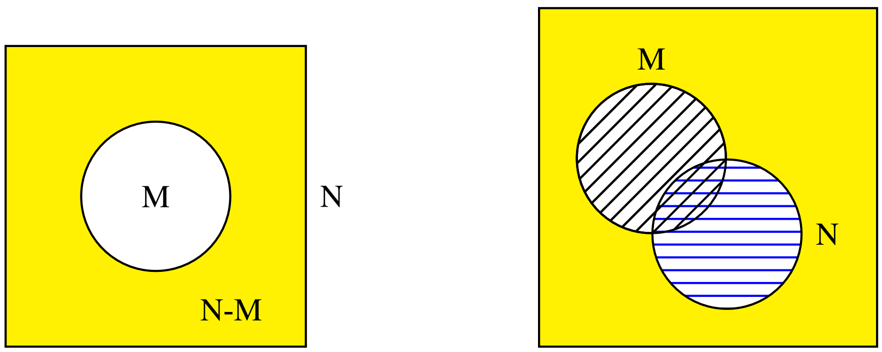
In der Abbildung (2.2)(a) ist der weiße Bereich der der Menge M, der gelbe Bereich der außerhalb der Menge M. Ist der Gesamtbereich ebenfalls eine Menge, hier N genannt, so ist der gelbe Bereich die Menge $N - M$. Sie umfasst alle Elemente, die in N aber nicht in M sind. In (2.2)(b) haben wir eine andere Konstellation. M und N sind hier nicht ineinander enthalten. M ist schräg schraffiert, N dagegen waagerecht. Wir haben nun die Bereiche $M \cap N$ (der doppelt schraffierte Bereiche) sowie $M \cup N$ (der nichtgelbe Bereich).

Es gelten die folgenden Gesetze.

(2.2) $M \cap \varnothing = \varnothing$ $\qquad$ $M \cup \varnothing = M$

Abbildung 2.2: Schnitt, Vereinigung und Komplement

(a) Komplement (b) Schnitt zweier Mengen



$$M \cap N = N \cap M \qquad M \cup N = N \cup M \tag{2.3}$$

$$M \cap (N \cap P) = (M \cap N) \cap P \qquad M \cup (N \cup P) = (M \cup N) \cup P \tag{2.4}$$

$$M \cap M = M \qquad M \cup M = M \tag{2.5}$$

Wir sagen auch, Schnitt ($\cap$) und Vereinigung ($\cup$) seien **kommutativ**, weil (2.3) gilt; sie seinen **assoziativ**, weil (2.4) gilt; sie seien **idempotent**, weil (2.5) gilt. Ich will ein paar Gesetze herausgreifen und plausibel machen. Zum Beispiel will ich zeigen, dass $M \cap N = N \cap M$ ist. Das machen wir mit Hilfe von Definition 1.1. Es sei $m \in M \cap N$. Nach Definition erfüllt x die Eigenschaft "$\_\_ \in M$ und $\_\_N$". Das bedeutet, das sowohl $x \in M$ als auch $x \in N$. Also ist $x \in N$ und es ist $x \in M$, das heißt, x erfüllt "$\_\_ \in N$ und $\_\_ \in M$", woraus wir nach Definition wiederum schließen, dass $x \in N \cap M$. Die Umkehrung geht genauso (man tausche einfach M und N füreinander aus). Ähnlich zeige ich, dass $M \cap M = M$. Dazu sei $x \in M$. Dann erfüllt x auch die Eigenschaft "$\_\_ \in M$ und $\_\_ \in M$", das heißt, $x \in M \cap M$. Falls umgekehrt $x \in M \cap M$, so erfüllt x die Eigenschaft "$\_\_ \in M$ und $\_\_ \in M$", also ist $x \in M$ und $x \in M$. Und so ist eben auch $x \in M$.

Die Gesetze mit der leeren Menge machen erfahrungsgemäß Schwierigkeiten. Deswegen möchte ich auch sie hier erläutern. Ist M eine Menge, so ist $M \cap \varnothing$ die Menge aller Gegenstände, die sowohl in M sind wie auch in der leeren Menge. Aber die leere Menge enthält nichts; also gibt es keinen Gegenstand, der in beiden Menge enthalten ist. Die Menge ist also leer: $M \cap \varnothing = \varnothing$. $M \cup \varnothing$ ist wiederum die

Menge der Gegenstände, die in M sind oder in $\varnothing$. Ein Gegenstand, der darin ist, muss aber in M sein, weil er nicht in $\varnothing$ sein kann. Also ist $M \cup \varnothing = M$.

Man mache sich klar, dass die Operation "–" weder kommutativ noch assoziativ ist und auch nicht idempotent. Dies bedeutet im Einzelnen:

① Im Allgemeinen ist $M - N \neq N - M$. Als Beispiel nenne ich $\{\spadesuit, \heartsuit\} - \{\heartsuit\} \neq \{\heartsuit\} - \{\spadesuit, \heartsuit\}$.

② Im Allgemeinen ist $M - (N - O) \neq (M - N) - O$. Als Beispiel nenne ich $\{\spadesuit, \heartsuit\} - (\{\heartsuit\} - \{\spadesuit\}) \neq (\{\spadesuit, \heartsuit\} - \{\heartsuit\}) - \{\spadesuit\}$.

③ Im Allgemeinen ist $M - M \neq M$. Denn $\{\spadesuit, \heartsuit\} - \{\spadesuit, \heartsuit\} \neq \{\spadesuit, \heartsuit\}$.

Ich sage "im Allgemeinen", weil es durchaus positive Beispiele geben kann. So ist, wenn etwa $M = N$ gilt, durchaus $M - N = N - M$. Diese Menge ist übrigens die leere Menge, dh $M - M = \varnothing$. Manchmal geht es halt, manchmal nicht. So gilt zum Beispiel auch $3 + (5 \times 1) = (3 + 5) \times 1$, aber deswegen ist im Allgemeinen nicht $x + y \times z = (x + y) \times z$.

Es gilt aber Folgendes.

$$M - \varnothing = M \qquad M - M = \varnothing \tag{2.6}$$

Außer den Gesetzen, die nur eine Operation enthalten, gibt es noch solche für zwei Operationen:

$$\begin{aligned} (M \cap N) \cup P &= (M \cup P) \cap (N \cup P) \\ (M \cup N) \cap P &= (M \cap P) \cup (N \cap P) \end{aligned} \tag{2.7}$$

Diese heißen **Distributivgesetze**. (Ein ähnliches Gesetz gilt auch für Zahlen: wir haben $(x + y) \cdot z = (x \cdot z) + (y \cdot z)$, nur dass ein analoges Gesetz für den Ausdruck $(x \cdot y) + z$ nicht existiert.)

Ferner haben wir für das relative Komplement folgende Rechenregel.

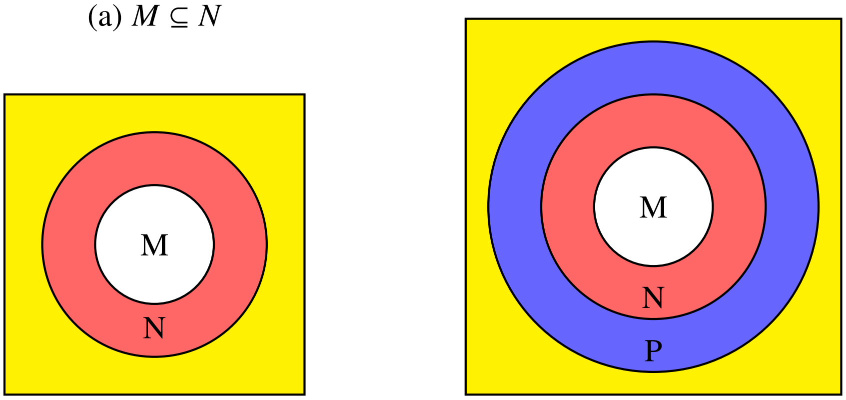
$$\begin{aligned} M \cap (N - O) &= (M \cap N) - O \\ M \cup (N - O) &= (M \cup N) - (O - M) \end{aligned} \tag{2.8}$$

Sehen wir uns an, warum das so ist. Es ist $x \in M \cap (N - O)$, wenn $x \in M$, $x \in N$ aber $x \notin O$. Dann ist aber auch $x \in M \cap N$ und $x \notin O$, also $x \in (M \cap N) - O$. Diese Argumentation kann man auch umkehren. Das zeigt die erste Gleichung. Nun zur zweiten. Ist $x \in M \cup (N - O)$, so ist $x \in M$ oder sowohl $x \in N$ als auch

Abbildung 2.3: Teilmengen

(b) $M \subseteq N \subseteq P$

(a) $M \subseteq N$



$x \notin O$. Also ist (a) $x \in M$ oder $x \in N$ und (b) $x \in M$ oder $x \notin O$. Also ist (a) $x \in M \cup N$ und (b) nicht ($x \notin M$ und $x \in O$), also $x \notin M - O$. Dies bedeutet zusammen $x \in (M \cup N) - (O - M)$. Auch diese Argumentation ist umkehrbar, und so gilt die zweite Gleichung.

Definition 2.1 (Teilmenge) *Wir sagen, M sei eine **Teilmenge** von N, in Zeichen $M \subseteq N$, wenn jedes Element von M auch Element von N ist. Symbolisch*

(2.9) $$M \subseteq N \leftrightarrow (\forall m)(m \in M \rightarrow m \in N)$$

*M ist **echte Teilmenge** von N, in Zeichen $M \subsetneq N$, wenn M Teilmenge von N ist, aber von N verschieden ist.*

Es gilt unter anderem stets

(2.10) $$\varnothing \subseteq M \quad M \subseteq M$$

Denn da kein Element in $\varnothing$ ist, ist offenbar jedes Element von $\varnothing$ auch schon in M.

Proposition 2.2 *Für zwei Mengen M und N ist genau dann $M = N$, wenn sowohl $M \subseteq N$ als auch $N \subseteq M$.*

Dies leuchtet unmittelbar ein. Ist $M = N$, so ist ja wegen $M \subseteq M$ auch $M \subseteq N$ und $N \subseteq M$ (man ersetze ein M durch N). Ist umgekehrt $M \subseteq N$, so ist jedes Element von M auch Element von N, und ist $N \subseteq M$, so ist jedes Element von N auch Element von M, und somit ist $M = N$.

Dieser Satz, so einfach er ist, leistet oft nützliche Dienste, wenn man zeigen möchte, dass zwei Mengen gleich sind.

Proposition 2.3 *Ist $M \subseteq N$ und $N \subseteq P$, so ist auch $M \subseteq P$.*

Denn sei x beliebig gewählt. Ist $x \in M$, so ist nach Voraussetzung auch $x \in N$. Und damit, wiederum nach Voraussetzung, ist $x \in P$. x war beliebig. Also ist $M \subseteq P$. Wiederum können die Euler-Venn Diagramme bei der Visualisierung nützen. Man symbolisiert die Tatsache, dass M Teilmenge von N ist dadurch, dass das Gebiet von M in dem Gebiet von N voll enthalten ist (Abbildung 2.3(a)). Ist zusätzlich $N \subseteq P$, so ist das Gebiet von N ganz in dem Gebiet von P enthalten, also das Gebiet von M in dem von P (b).

Ohne Begründung gebe ich noch ein paar weitere nützliche Tatsachen bekannt.

Proposition 2.4 *Es gilt:*

① $M \cap N \subseteq M$, $M \cap N \subseteq N$.

② *Ist $P \subseteq M$ und $P \subseteq N$, so ist $P \subseteq M \cap N$.*

③ $M \subseteq M \cup N$, $N \subseteq M \cup N$.

④ *Ist $M \subseteq P$ und $N \subseteq P$, so ist $M \cup N \subseteq P$.*

Laut Proposition 2.2 ist $M = N$ genau dann der Fall, wenn sowohl $M \subseteq N$ als auch $N \subseteq M$. Man unterscheide sorgsam zwischen $M \in N$ und $M \subseteq N$. Das erste sagt, dass M ein Element von N ist, dass zweite, dass jedes Element von M ein Element von N ist. Es kann nicht zugleich $M \in N$ und $M \subseteq N$ sein. (Dies ist nicht ganz einfach zu zeigen.) Ich weise aber hier ausdrücklich darauf hin, dass dies nicht deswegen gilt, weil es einen Unterschied gibt zwischen Elementen und Mengen. Den gibt es nicht! Eine Menge kann auch Element einer anderen Menge sein. Deswegen soll man den Gebrauch von Kleinbuchstaben (m) für Elemente von Mengen nicht so verstehen, dass hier über ganz andere Objekte geredet wird. Die Mengenlehre kennt nur eine Sorte Gegenstand. Allerdings ist es so, dass einige Dinge, wir nannten sie die ersten Gegenstände, besonders sind, weil sie zwar Elemente sind, aber nicht Elemente enthalten. Diese heißen auch Urelemente. Die formale Mengenlehre schafft es, gänzlich ohne sie auszukommen (also Mengen gleichsam aus dem Nichts zu schaffen), aber das ist in der Praxis zu mühsam. Für Urelemente gilt denn auch nicht das Extensionalitätsprinzip: denn Urelemente enthalten nach Definition keinerlei Elemente, müssten dann also gleich der leeren Menge sein.

Es sei nun X eine Menge und $M \subseteq X$. Dann heißt $X - M$ auch das **relative Komplement** von M in X. Dies wird auch manchmal mit $-_X M$ bezeichnet. Für diese Operation gilt

$$(2.11) \qquad -_X -_X M = M$$

Anders geschrieben:

$$(2.12) \qquad X - (X - M) = M$$

Dazu überlege man sich, dass genau dann $x \in X - (X - M)$, wenn $x \in X$ und nicht $x \in X - M$. Das letztere ist genau dann der Fall, wenn $x \in X$ und nicht ($x \in X$ und $x \notin M$). Dies ist gleichwertig mit $x \in X$ und ($x \notin X$ oder $x \in M$). Und dies wiederum ist gleichwertig mit $x \in X$ und $x \in M$; dies ist dasselbe wie $x \in M$, weil $M \subseteq X$. Die folgenden Gesetze sind bekannt als **de Morgan'sche Gesetze**.

$$(2.13) \qquad -_X(M \cap N) = (-_X M) \cup (-_X N) \qquad -_X(M \cup N) = (-_X M) \cap (-_X N)$$

(Diese gelten auch, wenn M und N nicht Teilmengen von X sind.) Sei dazu $x \in -_X(M \cap N)$. Das heißt: $x \in X$ und $x \notin M \cap N$. Das wiederum heißt: $x \in X$ und ($x \notin M$ oder $x \notin N$), also $x \in X$ und $x \notin M$ oder $x \in X$ und $x \notin N$. Das ist nichts anderes als $x \in (-_X M) \cup (-_X N)$. Das andere Gesetz zeigt man analog.

Am Schluss noch zwei weitere nützliche Gesetze, die sogenannten Absorptionsgesetze:

$$(2.14) \qquad (M \cap N) \cup N = N \qquad (M \cup N) \cap N = N$$

Diese sieht man am besten so. Da $M \cap N \subseteq N$, so ist $(M \cap N) \cup N = N$, wegen Proposition 2.4, ①. Da $N \subseteq M \cup N$, so ist $(M \cup N) \cap N = N$, wegen Proposition 2.4, ③.

Kommen wir nun zu einer weiteren Operation.

Definition 2.5 (Potenzmenge) *Es sei M eine Menge. Die Potenzmenge von M, in Zeichen $\wp(M)$, $P(M)$ oder $\mathrm{pow}(M)$, ist diejenige Menge, welche sämtliche Teilmengen von M (und nur diese) enthält.*

Machen wir uns das an einem Beispiel klar. Nehmen wir die Menge $M := \{\diamondsuit, \heartsuit, \spadesuit, \clubsuit\}$. In Tabelle 1.1 habe ich sämtliche Teilmengen der Menge M aufgelistet, insgesamt 16 Stück. Die Potenzmenge $\wp(M)$ enthält nun alle diese 16 Mengen.

Ich gebe noch ein paar mehr Beispiele.

- $\wp(\{\mathrm{a}, \mathrm{b}) = \{\varnothing, \{\mathrm{a}\}, \{\mathrm{b}\}, \{\mathrm{a}, \mathrm{b}\}\}$.
- $\wp(\{\Box\}) = \{\varnothing, \{\Box\}\}$.
- $\wp(\varnothing) = \{\varnothing\}$. Das liegt daran, dass die leere Menge Teilmenge der leeren Menge ist (und auch nur sie). Also enthält die Potenzmenge der leeren Menge die leere Menge, ist aber selbst deswegen nicht leer.

Es ist durchaus gestattet, von einer Potenzmenge wiederum die Potenzmenge zu bilden. So haben wir

$$\wp(\wp(\{\Box\})) = \{\varnothing, \{\varnothing\}, \{\{\Box\}\}, \{\varnothing, \{\Box\}\}\} \tag{2.15}$$

Allerdings werden diese Mengen recht schnell sehr groß.

Satz 2.6 *Es habe M n Elemente. Dann hat $\wp(M)$ 2^n Elemente.*

Hat also M 3 Elemente, so hat $\wp(M)$ schon $2^3 = 8$, $\wp(\wp(M))$ $2^8 = 256$ Elemente, und $\wp(\wp(\wp(M)))$ 2^{256} Elemente, mehr als 10^{75}, eine 1 mit 75 Nullen dahinter, also eine wahrhaft riesige Zahl!

Übungen

[5] Hier ein Rezept, um die Vereinigung zweier endlicher Mengen zu berechnen. Gegeben seien die Mengen M und N. Wir schreiben zuerst die Elemente von M auf und dann die Elemente von N. Ist dieses Rezept richtig? Wo könnte ein Problem sein? Können wir alternativ auch erst die Elemente von N aufschreiben und dann die von M?

[6] Noch einmal nehmen wir die Menge der Primzahlen < 20. Berechnen Sie:

- das Komplement der Menge der Primzahlen mit gerader Quersumme.
- die Vereinigung der Menge der Primzahlen > 10 mit der Menge der geraden Primzahlen.
- den Schnitt aus der Menge der Primzahlen > 10 mit der Menge der Primzahlen x, für die $x^2 < 200$.

[7] Ich behaupte, dass $M \subseteq N$ genau dann gilt, wenn $M \cap N = M$. Versuchen Sie zu zeigen, dass das richtig ist. *Anleitung.* In der Vorlesung wurde gezeigt, dass $M \cap N = M$ dasselbe bedeutet, wie $M \cap N \subseteq M$ und $M \subseteq M \cap N$.

[8] Vor uns liegt eine Tüte mit Bausteinen. Dies sei unser Grundbereich. Wir bilden die Menge der Ecksteine, die Menge der roten Bausteine und schließlich die Menge der roten Ecksteine. Wie bekommen wir die dritte Menge aus den beiden ersten?

Man vereinfache die folgenden Ausdrücke:

1. $(A \cup C) \cap (B - A)$
2. $(Z \cup U) \cap (U \cap Y)$

[9] Zeigen Sie: *Es ist* $(M \cap N) \cup P = M \cap (N \cup P)$*, sofern* $P \subseteq M$. Dazu können Sie entweder mit Umformungen arbeiten oder mit Euler-Venn-Diagrammen.

Kapitel 3

Relationen

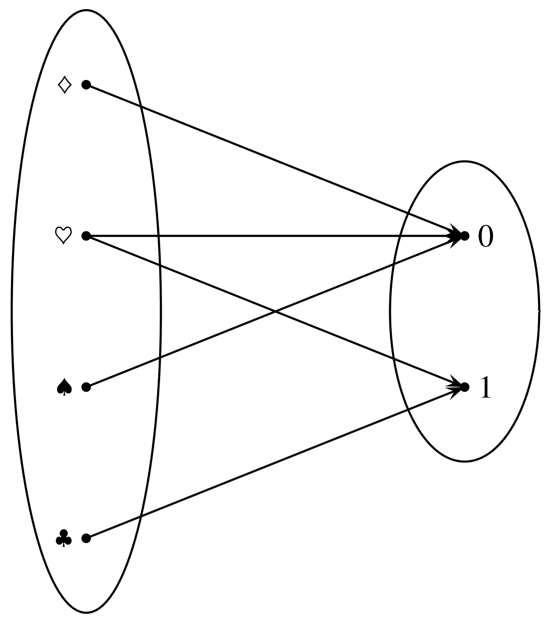
Es sei M eine Menge. Eine Teilmenge von M ist eindeutig dadurch bestimmt, dass man weiß, welches Element von M zu ihr gehört und welches nicht. Es sei nun N eine weitere Menge. Eine *Relation von M nach N* ist nun eindeutig dadurch bestimmt, auf welches Paar von Elementen von M und N diese Relation zutrifft. Genauso wie bei Mengen kann man das in einer Tabelle veranschaulichen. Sei $M := \{\heartsuit, \clubsuit, \spadesuit, \diamondsuit\}$ und $N := \{0, 1\}$. In der folgenden Tabelle sind drei Relationen, R_1, R_2 und R_3 gegeben.

(3.1)

M	N	R_1	R_2	R_3
$\diamondsuit$	0	+	−	+
$\diamondsuit$	1	−	+	−
$\heartsuit$	0	+	−	+
$\heartsuit$	1	+	−	+
$\spadesuit$	0	+	+	+
$\spadesuit$	1	−	−	−
$\clubsuit$	0	−	−	−
$\clubsuit$	1	+	+	+

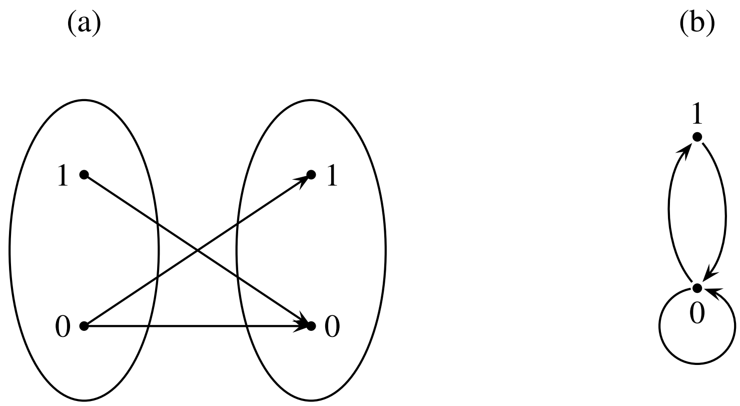
Dabei steht in der Spalte für R_1 ein +, falls die jeweiligen Elemente in Relation R_1 stehen. Da in der ersten Zeile ein + steht, so sagt dies, dass $\diamondsuit$ in der Relation R_1 zu 0 steht, wofür wir auch kurz „$\diamondsuit\, R_1\, 0$” schreiben. $\diamondsuit$ steht aber nicht in Relation R_2 zu 0, da in der Spalte für R_2 ein Minuszeichen steht. Wiederum steht $\diamondsuit$ in Relation R_2 zu 1, aber nicht in Relation R_1. Und so weiter. Da x genau dann in Relation R_1 zu y steht, wenn es auch in Relation R_3 zu y steht, ist $R_1 = R_3$. Aber es ist $R_1 \neq R_2$, weil einerseits $\diamondsuit\, R_1\, 0$ aber andererseits nicht $\diamondsuit\, R_2\, 0$. Eine weitere Möglichkeit der Veranschaulichung sind Pfeildiagramme. Dabei zeichnen wir zwei Gebiete für M

Abbildung 3.1: Die Relation R_1



und N und tragen einen Pfeil von m nach n ein, wenn m in Relation zu n steht. Ein Beispiel ist in Abbildung 3.1 gegeben. Sie veranschaulicht die Relation R_1 (wie auch R_3, da diese ja gleich sind). Dabei ist es wichtig zu unterscheiden, welches Element zuerst genannt wird. Es steht also $\diamondsuit$ in Relation R_1 zu 0, und es steht nicht etwa 0 in Relation R_1 zu $\diamondsuit$. Wenn nun M und N keine gemeinsamen Elemente haben, so spielt die Reihenfolge der Spalten keine erkennbare Rolle. Ist aber zum Beispiel $M = N = \{0, 1\}$, so kann man nicht umhin, genau zwischen der ersten Menge und der zweiten zu unterscheiden. Diese dürfen nämlich sogar gleich sein. Dabei haben wir in diesem Fall zwei Möglichkeiten, unsere Relationen zu symbolisieren. Entweder malen wir die Menge zweimal auf, oder wir malen sie nur einmal auf und zeichnen die Relation als Pfeildiagramm auf der Menge. Betrach-

Abbildung 3.2: Die Relation $\{\langle 0,1\rangle, \langle 1,0\rangle, \langle 0,0\rangle\}$



ten wir nun die folgenden Relationen.

(3.2)

M	N	S_1	S_2	S_3
0	0	+	−	−
0	1	−	+	+
1	0	−	−	−
1	1	+	−	+

Wir haben jetzt $0\,S_2\,1$ aber nicht $1\,S_2\,0$. (Bei der Relation S_1 gibt es dagegen keinen Unterschied zwischen $x\,S_1\,y$ und $y\,S_1\,x$, egal, was wir für x und y einsetzen; S_1 ist, wie wir unten sagen werden, *symmetrisch*.) Das liegt daran, dass nach Konvention das linke Element zuerst genannt wird. Bei Relationen müssen wir also anders als bei Mengen zwischen erstem Element und zweitem Element unterscheiden. Wie aber können wir dies tun?

Die Definition der Relation lässt sich mit Hilfe von Mengen so umbauen, dass wir genau sehen können, was die Eigenschaften einer Relation sind, wenn wir nur begriffen haben, wie Mengen funktionieren. Zunächst aber definiere ich ohne Hilfe von Mengen, was ein geordnetes Paar sein soll.

Definition 3.1 (Geordnetes Paar) *Das (**geordnete**) **Paar bestehend aus** m **und** n wird mit $\langle m,n\rangle$ bezeichnet. Dabei ist m das **erste Element** und n das **zweite Element** des Paares. Genau dann ist $\langle m,n\rangle = \langle m',n'\rangle$, wenn $m = m'$ und $n = n'$.*

Man beachte, dass im Gegensatz zu Mengen $\langle m, n\rangle$ nicht immer gleich $\langle n, m\rangle$ ist. Dies ist nur dann der Fall, wenn $m = n$. Außerdem ist $\langle m, n\rangle$ stets verschieden von m und n. Und drittens gibt es eine Funktion, die von einem Paar das erste Element und das zweite Element findet. Wir nennen diese Funktionen auch **Projektionen**. Diese sind

(3.3) $$\pi_1(\langle m, n\rangle) := m, \qquad \pi_2(\langle m, n\rangle) := n$$

Es gibt nun eine Möglichkeit, Paare als Mengen darzustellen, nämlich so.

(3.4) $$\langle m, n\rangle := \{m, \{n, m\}\}$$

Dies ist dann auch unsere endgültige Definition eines geordneten Paares. Dass dies eine gute Wahl ist, zeigt der folgende Satz. Er bestätigt, dass die Setzung in (3.4) den Anforderungen aus Definition 3.1 genügt.

Proposition 3.2 *Es gilt*

1. $\{m, \{n, m\}\} = \{m', \{n', m'\}\}$ *genau dann, wenn* $m = m'$ *und* $n = n'$.
2. $m \neq \{n, \{m, n\}\} \neq n$.

Beweis. Zunächst zur ersten Behauptung. Ist $m = m'$ und $n = n'$, dann ist sicher $\{m, \{n, m\}\} = \{m', \{n', m'\}\}$. Umgekehrt, sei $\{m, \{n, m\}\} = \{m', \{n', m'\}\}$. Dann treten zwei Fälle ein. (a) $m = m'$ oder (b) $m = \{n', m'\}$. Im Fall (a) ist dann $\{n, m\} = \{n', m'\}$, weil ja die Menge dieselben Elemente enthalten. Die ersten beiden Elemente sind gleich, so müssen auch die zweiten einander gleich sein. Also $\{m, n\} = \{m', n'\}$. Und weil dann $m = m'$, so ist nach demselben Argument $n = n'$. Im Fall (b) ist zusätzlich noch $\{n, m\} = m'$, und so $m = \{n', m'\} = \{n', \{n, m\}\}$. Das widerspricht aber der Eigenschaft von Mengen, dass sie nicht sich selbst (direkt oder indirekt) enthalten können (Fundierungseigenschaft). Nun zur zweiten Behauptung. Falls $n = \{n, \{m, n\}\}$, so $n \in n$, wiederum ein Widerspruch zur Fundierungeigenschaft. Ebenso $m = \{n, \{m, n\}\}$ führt zu $m \in \{m, n\} \in \{m, \{n, m\}\} = m$, und auch dies widerspricht der Fundierungseigenschaft. ⊣

Ich hebe noch einmal das kombinatorische Argument hervor. Es seien $P = \{a, b\}$ und $Q = \{c, d\}$ zwei Mengen und $P = Q$. Dann gilt (a) $a = c$ und $b = d$, oder (b) $a = d$ und $b = c$. Denn zunächst einmal haben wir $a \in P$, sodass also $a \in Q$ sein muss (Definition 1.1). Also ist $a = c$ oder $a = d$. Da nun auch $b \in P$, so muss ebenfalls $b = c$ oder $b = d$. Von den vier möglichen Kombinationen bleiben nur 2. Denn ist $a = c$ und $b = c$, so muss *zusätzlich* $c = d$ sein. Denn aus $d \in Q$ folgt ja auch $d \in P$, sodass $d = a$ oder $d = b$ sein muss. In beiden Fällen ist dann $d = c$. Dies erzwingt $a = b = c = d$, was aber sowohl durch den Falle (a) wie auch den Fall (b) abgedeckt ist. Ebenso sieht man, dass $a = d$ und $b = d$ zur Folge hat, dass $a = b = c = d$.

Ich füge noch hinzu, dass $\langle m, n\rangle$ stets zwei Elemente hat, nämlich m und $\{m, n\}$. Diese sind deswegen verschieden, weil $m \in \{m, n\}$ und deswegen nicht $m = \{m, n\}$ sein kann.

Wie definieren wir nun die Projektionen? Wie können wir praktisch aus einem Paar die Komponenten herausfischen? Dazu betrachte ich zwei Fälle. Der erste Fall ist, dass $\langle m, n\rangle = \{m, \{n, m\}\}$ ist, wo $m = n$. In diesem Fall ist $\langle m, n\rangle = \langle m, m\rangle = \{m, \{m\}\}$. Unser Paar hat dann zwei Elemente, nennen wir sie u und v, von denen eines die Eigenschaft hat, das andere als einziges Element zu enthalten. Ohne Einschränkung ist $u = \{v\}$. Dann ist $v = m$. Denn wäre $U = m$, so hätten wir ja wieder eine unfundierte Menge! Somit können wir m wiederfinden, und gleichzeitig können wir erkennen, dass $m = n$ ist.

Nun also der Fall, wo $m \neq n$. Dann hat $\{m, \{n, m\}\}$ wiederum zwei Elemente U und V, wobei ohne Einschränkung $U \in V$ (aber dann nicht $V \in U$). In diesem Fall ist $U = m$, und dann ist $V = \{m, n\}$ für ein n, das sich eindeutig wiederfinden lässt.

Definition 3.3 (Projektion) *Die Funktionen π_1 und π_2 sind auf allen Mengen der Form $p = \{m, \{m, n\}\}$ definiert. p hat genau zwei Elemente, das heißt $p = \{u, v\}$, wobei ohne Beschränkung der Allgemeinheit $u \in v$. Dann ist $\pi_1(p) := u$. $\pi_2(p)$ ist wie folgt definiert. Ist $v = \{u\}$, so ist $\pi_2(p) := u$, ansonsten ist $\pi_2(p) = w$, wo w dasjenige Element ist mit $v = \{u, w\}$.*

All dies dient nur der Zusicherung, dass unsere Definition all das tut, was sie auch tun soll.

Definition 3.4 (Kartesisches Produkt, Relation) *Es ist $M \times N := \{\langle m, n\rangle : m \in M, n \in N\}$. Diese Menge heißt das* ***kartesische Produkt*** *von M und N. Eine* ***Relation von*** *M* ***nach*** *N ist eine beliebige Teilmenge R von $M \times N$. Im Fall, wo $M = N$, sagen wir auch, R sei eine Relation* ***auf*** *M. Wir sagen, x* ***steht in Relation*** *R* ***zu*** *y, falls $\langle x, y\rangle \in R$. Wir schreiben alternativ auch „$x\,R\,y$”.*

Man bemerke, dass Relationen jetzt auch als besondere Mengen dargestellt sind. Insofern können wir jetzt die gleichen Operationen auf sie anwenden.

Wir besitzen jetzt auch ein Kriterium, wann zwei Relationen gleich sind. Es gilt $R = S$ genau dann, wenn für alle x und y gilt $\langle x, y\rangle \in R$ genau dann, wenn

$\langle x, y\rangle \in S$. Ist nun $R \subseteq M \times N$ und $S \subseteq M' \times N'$, so kann auch dann $R = S$ gelten, wenn $M \neq M'$ oder $N \neq N'$. Insbesondere gilt: sind M, M', N, N' Mengen mit $M \subseteq M'$ und $N \subseteq N'$, so ist jede Relation von M nach N auch eine Relation von M' nach N'. Als Beispiel sei nur die leere Relation erwähnt. Wenn $R \subseteq M \times N$ und $S \subseteq M' \times N'$, und ist für jedes Paar $\langle x, y\rangle \in S$ auch $\langle x, y\rangle \in R$, so ist bereits $S = R$. Die Elemente, um die sich M und M' bzw. N und N' unterscheiden, spielen also keine Rolle. Deswegen ist in diesem Fall $S \subseteq M \times N$.

Nach dieser Definition sind $\varnothing$ und $M \times N$ zum Beispiel Relationen von M nach N. Und mit R und S sind auch $R \cup S$, $R \cap S$ sowie $M \times N - R$ Relationen von M nach N. Dazu ein paar Beispiele. Sei V die Relation „ist Vater von", A und die Relation „ist älter als", L „ist Lehrer von". Das soll bedeuten, dass $x\, V\, y$ (also $\langle x, y\rangle \in V$) genau dann gilt, wenn x Vater von y ist. Oder anders

(3.5) $\qquad V = \{\langle x, y\rangle : x \text{ ist Vater von } y\}$

Dann ist $V \cap L$ die Relation „ist Vater von und Lehrer von" (oder auch „ist Vater und Lehrer von"). Zum Beispiel war Leopold Mozart der Vater und Lehrer von Wolfgang Amadeus Mozart. Denn es ist x Vater und Lehrer von y, wenn x Vater von y ist und x Lehrer von y ist, was nichts anderes bedeutet als $x\, V\, y$ und $x\, L\, y$, und wieder $x\, (V \cap L)\, y$. (Letzteres ist kurz für $\langle x, y\rangle \in V \cap L$.) Man lasse sich also nicht verwirren von der Tatsache, dass es Relationen sind; Relationen sind Mengen, und auf Mengen sind Schnitt und Vereinigung bereits definiert. So ist $V \cup L$ die Relation „ist Vater oder Lehrer von". Es gilt übrigens $A \subseteq V$, weil Väter nun einmal älter sind als ihre Kinder. Und deswegen ist $A \cap V = V$ („ist Vater von und älter als" ist nichts anderes als „ist Vater von"). Und so weiter.

Es gibt außer diesen auch noch andere Operationen. Definiere zum Beispiel

(3.6) $\qquad R^{\smile} := \{\langle n, m\rangle : \langle m, n\rangle \in M \times N\}$

Es heißt $R^{\smile}$ die zu R **konverse Relation**. So ist zum Beispiel „links von" konvers zu „rechts von". Denn wenn a links von b, so ist b rechts von a und umgekehrt. Die Relation „ist Vorfahre von" ist konvers zu „ist Nachfahre von".

Eine weitere wichtige Operation ist das *Relationenprodukt*. Es sei $R \subseteq M \times N$ und $S \subseteq N \times P$. Dann setze

(3.7) $\qquad R \circ S = \{\langle x, z\rangle : \text{es gibt ein } y \in N \text{ mit } x\, R\, y\, S\, z\}$

Es ist also $R \circ S$ das **Produkt** von R und S. Es kommt dabei auf die Reihenfolge an. Zwar kann man $S \circ R$ stets bilden, da es als Menge von Paaren durchaus definiert werden kann, auch wenn zunächst einmal die Definition verlangt, dass für geeignete Mengen M', N' und P' gilt, dass $S \subseteq M' \times N'$ und $R \subseteq N' \times P'$. Diese kann man aber durchaus finden (etwa setzen wir $M' := N' := P' := M \cup N \cup P$). Aber trotzdem ist in der Regel $S \circ R \neq R \circ S$.

In den natürlichen Sprachen taucht das Relationenprodukt unter anderem bei komplexen Nominalphrasen mit Relationalnomina auf. So bezeichnet „das Auto des Nachbarn" eine Relation, die zwischen x und z besteht, falls es ein y gibt, sodass x das Auto von y ist und y der Nachbar von z. Symbolisch: bezeichnet A die Relation „Auto von" und N die Relation „Nachbar von", so bezeichnet $A \circ N$ die Relation „Auto des Nachbarn von".

Definition 3.5 (Transitivität, Symmetrie, usf.) *Es sei $R \subseteq M \times M$ eine Relation. R heißt **transitiv**, falls aus $x\,R\,y$ und $y\,R\,z$ folgt $x\,R\,z$. R ist **symmetrisch**, wenn aus $x\,R\,y$ folgt $y\,R\,x$. R heißt **reflexiv**, falls für alle $x \in M$ gilt $x\,R\,x$. Ist R reflexiv, symmetrisch und transitiv, so heißt R eine **Äquivalenzrelation**. R ist **linear**, falls R transitiv ist und falls für alle $x, y \in M$ $x\,R\,y$ oder $x = y$ oder $y\,R\,x$ gilt (aber nur eines von den dreien).*

Achtung. In manchen Büchern erlaubt man, dass lineare Relationen auch reflexiv sein dürfen. Dies ist hier explizit ausgeschlossen. Denn wenn $x = y$ gilt, darf nicht auch $x\,R\,y$ sein. Insbesondere ist deswegen für eine lineare Relation niemals $x\,R\,x$!

Eine Relation auf M ist **symmetrisch**, wenn sie gleich ihrer Konversen ist. Man sollte bei der Negation dieser Konzepte Vorsicht walten lassen. Eine Relation $R \subseteq M \times M$ ist schon dann nicht reflexiv, wenn es ein einziges x gibt, für das nicht $x\,R\,x$. Dagegen nennt man eine Relation **irreflexiv**, wenn für *kein* $x \in M$ gilt $x\,R\,x$. Ein Beispiel einer nicht reflexiven aber dennoch nicht irreflexiven Relation ist $\{\langle 0,0\rangle, \langle 0,1\rangle, \langle 1,0\rangle\} \subseteq \{0,1\} \times \{0,1\}$. Lineare Relationen sind also nicht nur nicht reflexiv, sie sind sogar irreflexiv. Analog für die Symmetrie. Eine Relation ist **asymmetrisch**, falls für alle $x, y \in M$ gilt: ist $x\,R\,y$, so ist nicht $y\,R\,x$. Eine asymmetrische Relation ist gewiss nicht symmetrisch, aber der Umgekehrte muss nicht gelten. Ein Beispiel dafür ist $\{\langle 0,0\rangle, \langle 0,1\rangle, \langle 1,1\rangle\} \subseteq \{0,1\} \times \{0,1\}$.

Nehmen wir ein paar Beispiele.

① Die Relation „ist Nachbar von" ist symmetrisch.

② Die Relation „$<$" auf den natürlichen Zahlen ist linear.

③ Die Relation „$\in$" ist nicht symmetrisch, nicht reflexiv, und nicht transitiv auf den Mengen. Genauer ist es so, dass niemals $x \in x$ ist („$\in$" ist irreflexiv), und falls $x \in y$, so ist nicht $y \in x$ („$\in$" ist asymmetrisch). Es kann aber sein, dass $x \in y \in z$ und auch $x \in z$. Ein Beispiel dafür liefern die Mengen $\underline{n}$, welche wir in Kapitel 5 besprechen.

Zum Abschluss noch ein weiteres Beispiel, wie wir mit Hilfe von Mengen recht wichtige komplexe Gegenstände definieren können.

Definition 3.6 (Partition) *Es sei M eine Menge. Eine **Partition** von M ist eine Menge $P \subseteq \wp(M)$ von nichtleeren Teilmengen von M derart, dass für alle $K, L \in P$ gilt $K = L$ oder $K \cap L = \varnothing$ und für jedes $x \in M$ ein $O \in P$ existiert derart, dass $x \in O$.*

Proposition 3.7 *Genau dann ist P eine Partition von M, wenn alle Mengen in P nichtleer sind und für jedes $x \in M$ genau ein $O \in P$ existiert mit $x \in O$.*

In einer Partition P sei $x \in K$ und $x \in L$. Dann ist entweder $K = L$ oder aber $K \cap L = \varnothing$. Der zweite Fall kann nicht auftreten; also ist x nur ein einer Partitionsmenge. Die Umkehrung ist ebenso leicht zu sehen.

Satz 3.8 *Es gibt genau so viele Partitionen über einer Menge M wie es Äquivalenzrelationen auf M gibt. Und zwar sei P eine Partition von M. Dann definiere R_P durch $x\,R_P\,y$ genau dann, wenn ein $O \in P$ existiert mit $x, y \in O$. Und sei R eine Äquivalenzrelation. Dann definiere $\Pi_R := \{R_x : x \in M\}$, wo $R_x := \{y : x\,R\,y\}$. Dann gilt*

① *$\Pi_{R_P} = P$ für jede Partition P von M.*

② *$R_{\Pi_A} = A$ für jede Äquivalenzrelation A auf M.*

Beweis. R_P ist sicher eine Äquivalenzrelation. Denn es ist $x \in O$ für ein $O \in P$ und so $x\,R_P\,x$. Ist nun $x\,R_P\,y$, so existiert ein $O \in P$ mit $x, y \in O$. Und dann ist aber auch $y\,R_P\,x$. Sei nun $x\,R_P\,y$ und $y\,R_P\,z$. Dann existiert ein $O \in P$ mit $x, y \in O$ und ein $O' \in P$ mit $y, z \in O'$. Da nun $y \in O'$ und $y \in O$, also $O \cap O' \neq \varnothing$, so ist $O = O'$. Daraus folgt $x\,R_P\,z$. Sei A eine Äquivalenzrelation. Dann ist Π_A eine Partition. Diese besteht aus den Mengen $A_x = \{y : x\,A\,y\}$. Zu jedem y ist $y \in A_y$, weil A reflexiv ist. Nun muss nur noch die Eindeutigkeit gezeigt werden. Sei $y \in A_x$ und $y \in A_z$. Dies bedeutet $x\,A\,y$ und $z\,A\,y$. Daraus folgt $y\,A\,z$ (A ist ja symmetrisch) und somit $x\,A\,z$ (A ist transitiv) und so $z \in A_x$. Ist nun $z\,A\,u$, so ist dann $x\,A\,u$, weswegen $A_z \subseteq A_x$ gilt. Ebenso sieht man aber auch ein, dass $A_x \subseteq A_z$ ist, und das bedeutet $A_x = A_z$.

Offensichtlich genügt es nunmehr, ① und ② zu zeigen. Es sei P eine Partition. Dann besteht Π_{R_P} aus den Mengen $R_{Px} := \{y : x\, R_P\, y\}$. Nach Definition ist $x\, R_P\, y$, falls es eine Menge $O \in P$ gibt mit $x, y \in O$. O ist durch x eindeutig bestimmt. Deswegen ist $R_{Px} = O \in P$. Also ist $\Pi_{R_P} \subseteq P$. Man sieht leicht, da Π_{R_P} eine Partition ist, dass beide gleich sein müssen.

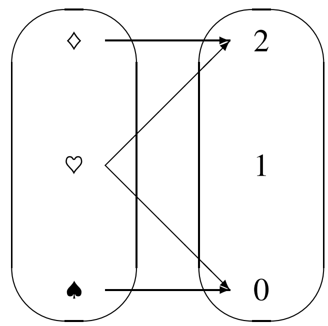
Nun sei A gegeben. Wir zeigen $R_{\Pi_A} = A$. Es ist $x\, R_{\Pi_A}\, y$ genau dann, wenn es ein $O \in \Pi_A$ gibt mit $x, y \in O$. Dies ist genau dann der Fall, wenn für ein z: $x \in A_z$ und $y \in A_z$, das heißt $z\, A\, x$ und $z\, A\, y$. Dies ist genau dann der Fall, wenn $x\, A\, y$. (Denn sei $z A \in x$ und $z\, A\, y$. Dann $x\, A\, z$ (Symmetrie) und so $x\, A\, y$ (Transitivität). Umgekehrt, falls $x\, A\, y$, so setze $z := x$. Dann ist $x\, A\, z$, also $z\, A\, x$, und $z\, A\, y$.) Dies beendet den Beweis. ⊣

Der Beweis benutzt ein Faktum, auf das ich im nächsten Kapitel noch eingehen werde: um zu zeigen, dass zwei Mengen A und B die gleiche Anzahl Elemente haben, geben wir eine Abbildung $f : A \to B$ und eine Abbildung $g : B \to A$ an derart, dass für alle $x \in A$ $g(f(x)) = x$ und für alle $y \in B$ $f(g(y)) = y$. (g heißt dann auch Umkehrfunktion von f.)

Übungen

[10] Betrachten Sie die Relationen „ist blutsverwandt mit" und ihre Negation (dh „ist nicht blutsverwandt mit") auf der Menge M der gegenwärtig lebenden Menschen. Welche dieser Relationen ist (a) symmetrisch, (b) reflexiv, (c) transitiv? *Anmerkung:* Was würde sich ändern, wenn wir stattdessen die Relationen auf der Menge aller Menschen betrachten, die gegenwärtig leben bwz. jemals gelebt haben? Sie können dabei selbstverständlich nur spekulieren. Fragen Sie sich etwa, was wäre, wenn die Schöpfungsgeschichte der Bibel wahr wäre.

[11] Verwandeln Sie die folgende Relation $R \subseteq \{a, b, c\} \times \{0, 2, 3\}$ in ein Pfeildiagramm: $\{\langle a, 0\rangle, \langle b, 2\rangle, \langle b, 3\rangle, \langle a, 3\rangle\}$. Schreiben Sie umgekehrt die folgende Relation als Menge auf:



[12] Es sei P die folgende Relation auf den natürlichen Zahlen: $x\,P\,y$ genau dann, wenn x und y von denselben Primzahlen geteilt werden. (Also ist $6\,P\,12$, weil ja 6 und 12 jeweils von 2 und 3 geteilt werden.) Ist dies eine Äquivalenzrelation?

[13] Es sei $A \subseteq M \times M$ symmetrisch. Ist dann das relative Komplement $B := M \times M - A$ auch symmetrisch?

Kapitel 4

Funktionen

Ein Relation R von M nach N heißt **Funktion** (von M nach N), wenn es zu jedem $x \in M$ genau ein $y \in N$ gibt mit $x\,R\,y$. Ich teile diese Definition in zwei Teile. Der erste Teil ist die Forderung, zu jedem x existiere *wenigstens* ein y mit $x\,R\,y$, und der zweite ist die Forderung, es existiere auch nur *höchstens* ein solches y. Die Definition ist es wert, sorgfältig studiert zu werden. Teil ② sagt nämlich sinngemäß dieses: je zwei Elemente, die zu x in Relation stehen sind gleich. (Natürlich handelt es sich dann nicht um *zwei* Elemente; die Formulierung unten vermeidet dieses Problem.)

Definition 4.1 (Funktion) *Es seien M und N zwei Mengen. Eine* ***Funktion*** *von M nach N, in Zeichen $f : M \to N$, ist eine Relation von M nach N mit folgenden Eigenschaften.*

① *Für jedes $x \in M$ existiert ein $y \in N$ mit $\langle x, y\rangle \in f$.*

② *Für jedes $x \in M$ und $y, y' \in N$ mit $\langle x, y\rangle \in f$ und $\langle x, y'\rangle \in f$ gilt $y = y'$.*

Gegeben x heißt das eindeutig bestimmte y mit $\langle x, y\rangle \in f$ der ***Wert von** f **an der Stelle*** *x und wird mit $f(x)$ bezeichnet.*

Man benutzt meist die Buchstaben f, g zum Bezeichnen von Funktionen. Statt zu sagen, dass x in Relation f zu y steht, sagen wir jetzt, f **ordnet** y x **zu** oder f **bildet** x **auf** y **ab**. In Zeichen schreiben wir auch $f : x \mapsto y$. Es sagt also $f : M \to N$, dass f eine Funktion von M nach N ist und $f : x \mapsto y$, dass die Funktion f das Element x auf das Element y abbildet. Ist $f : M \to N$, so heißt M der **Definitionsbereich** von f und N der **Wertebereich**. Der Definitionsbereich wird auch mit „dom f" oder „dom(f)" (von Englisch *domain*) bezeichnet.

Man beachte, dass Funktionen spezielle Relationen sind. Sei zum Beispiel $M = \{\heartsuit, \diamondsuit, \spadesuit, \clubsuit\}$ und $N = \{0, 1\}$, dann ist die folgende Menge eine Funktion von M nach N:

(4.1) $\{\langle\spadesuit, 1\rangle, \langle\clubsuit, 1\rangle, \langle\heartsuit, 0\rangle, \langle\diamondsuit, 0\rangle\}$

Man beachte, dass $\langle\spadesuit, 1\rangle$ kurz ist für $\{\spadesuit, \{\spadesuit, 1\}\}$. Insofern ist die vorliegende Menge schon ein recht kompliziertes Gebilde!

Wichtig ist also zweierlei: Funktionen ordnen jedem Element ihres Definitionsbereichs ein Element zu; und sie ordnen jedem Element des Definitionsbereichs auch nur ein einziges Element zu. Die folgenden Relationen sind deswegen keine Funktionen:

(4.2) $K := \{\langle\clubsuit, 1\rangle, \langle\heartsuit, 0\rangle, \langle\diamondsuit, 0\rangle\}$

(4.3) $L := \{\langle\spadesuit, 1\rangle, \langle\spadesuit, 0\rangle, \langle\clubsuit, 1\rangle, \langle\heartsuit, 0\rangle, \langle\diamondsuit, 0\rangle\}$

K hat eine Lücke: das Element $\spadesuit$ hat keinen Wert; L ordnet ihm hingegen gleich zwei Werte zu.

Die Definition von Funktionen hat zur Folge, dass wir ein eindeutiges Kriterium besitzen, wann zwei Funktionen gleich sind. Es ist $f = g$ genau dann, wenn für alle x im Definitionsbereich von f (der auch der Definitionsbereich von g ist) gilt $f(x) = g(x)$. Das bedeutet Folgendes. Ist $f : M \to N$ und $g : M' \to N'$, so ist $f = g$ genau dann, wenn $M = M'$, und wenn dann auch für alle $x \in M$ gilt $f(x) = g(x)$. Es muss *nicht* gelten, dass $N = N'$. Allerdings sind die Elemente, um die sich N und N' unterscheiden dürfen, nicht im Bild einer dieser Funktionen. Das Bild ist dabei wie folgt definiert.

Sei $f : M \to N$ gegeben und $K \subseteq M$. Dann ist

(4.4) $f[K] := \{f(y) : y \in K\}$

Wir nennen $f[K]$ das (**direkte**) **Bild von** K **unter** f. Das **Bild von** f (ohne Zusatz) ist definiert als $f[M]$. Es gilt $f[K \cup L] = f[K] \cup f[L]$ aber in der Regel nicht $f[K \cap L] = f[K] \cap f[L]$. (Denn falls $K = \{\spadesuit\}$ und $L = \{\clubsuit\}$ ist und $f(\spadesuit) = f(\clubsuit) = 1$, so ist $f[K \cap L] = f[\varnothing] = \varnothing$, jedoch ist $f[K] = \{1\} = f[L]$, weswegen $f[K] \cap f[L] = \{1\}$.)

Ich möchte an dieser Stelle noch auf eine Feinheit hinweisen. Wir reden zum Beispiel von der Nachfolgerfunktion auf den natürlichen Zahlen ($\mathbb{N}$) und von der Nachfolgerfunktion auf den ganzen Zahlen ($\mathbb{Z}$). Beide Funktionen ordnen der Zahl x die Zahl $x + 1$ zu. Trotzdem sind sie als Mengenobjekte verschieden; die Nachfolgerfunktion auf den ganzen Zahlen enthält zum Beispiel das Paar $\langle -1, 0\rangle$, die

Nachfolgerfunktion auf den natürlichen Zahlen nicht, weil ja -1 keine natürliche Zahl ist. Wir reden aber von den beiden als derselben Funktion, weil wir in diesem Fall nicht die Menge sondern das Zuordnungsprinzip meinen (dieses ist $x \mapsto x + 1$). Das Zuordnungsprinzip ist in der Tat identisch, insofern die Addition auf den verschiedenen Zahlbereichen sich nicht unterscheidet. Das bedeutet, dass zwar $x + 1$ in den natürlichen Zahlen nur dann erklärt ist, wenn x eine natürliche Zahl ist, aber das Ergebnis in den natürlichen Zahlen dasselbe ist wie in den ganzen Zahlen. Es ist $2 + 1 = 3$, egal, ob wir 2 als natürliche Zahl oder als ganze Zahl verstehen. Das ist auch der Grund, warum wir dasselbe Symbol schreiben, nämlich „+". Dass dies nicht ganz so selbstverständlich ist, sieht man, wenn man stattdessen die Vorgängerfunktion anschaut, also die Abbildung $x \mapsto x - 1$. Auf den ganzen Zahlen hat jede Zahl einen Vorgänger. Auf den natürlichen Zahlen aber gibt es zu 0 keinen Vorgänger, und so existiert die Vorgängerfunktion nicht auf den natürlichen Zahlen. Man kann die Lage „retten", indem man 0 zum Vorgänger von 0 erklärt, aber dann hat man eine *andere* Funktion auf $\mathbb{N}$ erklärt!

Sei $f : M \to N$. Und sei $K \subseteq M$. Dann können wir f auf K einschränken. Dies ist wie folgt definiert.

(4.5) $$f \restriction K := \{\langle x, y\rangle : \langle x, y\rangle \in f, x \in K\}$$

Eine alternative Definition dazu ist

(4.6) $$f \restriction K := f \cap (K \times N)$$

Sei zum Beispiel $s : \mathbb{Z} \to \mathbb{Z} : x \mapsto x + 1$ die Nachfolgerfunktion auf den ganzen Zahlen. Dann ist $s \restriction \mathbb{N} : \mathbb{N} \to \mathbb{Z}$ die Einschränkung auf die natürlichen Zahlen. Diese ist allerdings identisch mit der Funktion $s \restriction \mathbb{N} : \mathbb{N} \to \mathbb{N}$, weil ja für jedes $x \in \mathbb{N}$ der Nachfolger auch in $\mathbb{N}$ ist. Der Wertebereich ist also eigentlich ein zusätzlicher Parameter, den wir jedesmal mitdenken, wenn wir eine Funktion definieren. Es ist gewissermaßen ein „intendierte Zielmenge". Insofern bezeichnet die Sprechweise „$f : M \to N$" mehr als nur eine Funktion; sie bezeichnet eine Funktion zusammen mit einer Menge N, die mindestens den Wertebereich von f enthält aber durchaus größer sein kann.

Ich komme noch einmal auf Relationen zurück. Es seien M und N zwei Mengen. Dann sind folgende Funktionen definiert (die Projektionen, siehe (3.3)).

(4.7) $$\pi_1(\langle x, y\rangle) := x, \qquad \pi_2(\langle x, y\rangle) := y$$

Es ist $\pi_1 : M \times N \to M$ und $\pi_2 : M \times N \to N$. Ist $R \subseteq M \times N$, so heißt $\pi_1[R]$ der **Vorbereich** der Relation, und $\pi_2[R]$ der **Nachbereich**.

Im Falle einer Funktion gibt es eine Asymmetrie zwischen Definitions- und Wertebereich. Ist f alleine gegeben, so lässt sich der Definitionsbereich herauslesen. Dies ist die Menge $\pi_1[f]$. Dies ist die Menge aller x, denen f einen Wert zuweist. (Ist allerdings $M \neq \pi_1[f]$, so ist f keine Funktion; das kann man dann natürlich nicht herauslesen. Alles was wir herausfinden können ist, welche der möglichen Mengen der Definitionsbereich sein muss, wenn f eine Funktion ist.) Der Wertebereich lässt sich aber nicht eindeutig bestimmen. Denn ist $f : M \to N$ und $N \subseteq N'$ so ist auch $f : M \to N'$.

Definition 4.2 (Surjektivität, Injektivität) *Es sei $f : M \to N$ eine Funktion. f heißt **injektiv**, falls aus $x \neq y$ folgt $f(x) \neq f(y)$. f heißt **surjektiv**, falls für jedes $y \in N$ ein $x \in M$ existiert mit $f(x) = y$. Ist f sowohl injektiv wie surjektiv, so heißt f **bijektiv**.*

Die Nachfolgerfunktion $f : x \mapsto x+1$ ist injektiv auf den natürlichen Zahlen. Denn wenn $x \neq y$, so ist auch $x+1 \neq y+1$. Aber sie ist nicht bijektiv, denn 0 ist nicht das Bild einer natürlichen Zahl (deswegen hat 0 keinen Vorgänger). Sie ist andererseits bijektiv auf der Menge der ganzen Zahlen. Die Funktion $f : x \mapsto x^2$ ist bijektiv auf der Menge der positiven reellen Zahlen; aber auf der Menge der reellen Zahlen ist sie weder injektiv noch surjektiv (es gibt keine Zahl i mit $i^2 = -1$; und es gibt zwei Zahlen x mit $x^2 = 4$). Auf der Menge der komplexen Zahlen schließlich ist diese Funktion surjektiv aber nicht injektiv (es existiert eine Quadratwurzel aus jeder Zahl, aber meist existieren zwei verschiedene Wurzeln).

Schauen wir noch einmal die Definition 4.1 an. Sei die Bedingung ① erfüllt. Dann gilt, dass $f^{\smile}$ surjektiv ist; denn das ist nichts anderes, als dass jedes $x \in M$ ein $y \in N$ findet mit $\langle y, x\rangle \in f^{\smile}$ (also $\langle x, y\rangle \in f$). Sei die Bedingung ② erfüllt. Dann ist $f^{\smile}$ injektiv. Denn das bedeutet ja, dass zu jedem $y \in N$ höchstens ein $x \in M$ existiert mit $\langle y, x\rangle \in f^{\smile}$ (dh. $\langle x, y\rangle \in f$). Und so haben wir noch, dass $f^{\smile}$ genau dann eine Funktion ist, wenn f bijektiv ist. Und dann ist $f^{\smile}$ auch bijektiv.

Zu diesem Sachverhalt gibt es noch einen etwas anderen Zugang. Ist $f : M \to N$ und $K \subseteq N$, so schreiben wir

$$f^{-1}[K] := \{x \in M : f(x) \in K\} \tag{4.8}$$

Selbst wenn K nur aus einem Element besteht, kann $f^{-1}[K]$ sehr groß sein. Sei zum Beispiel $f : \mathbb{N} \to \mathbb{N}$ definiert durch $f(n) := 0$ wenn n gerade und $f(n) := 1$ sonst. Dann ist $f^{-1}(\{0\})$ die Menge aller geraden Zahlen. Es kann auch passieren, dass $f^{-1}[K]$ leer ist, wie zum Beispiel $f^{-1}(\{2\})$.

Proposition 4.3 *Sei $f : M \to N$. Genau dann ist f surjektiv, wenn aus $K \subseteq N$ und $K \neq \varnothing$ folgt $f^{-1}[K] \neq \varnothing$. Genau dann ist f injektiv, wenn $f^{-1}(\{x\})$ stets höchstens ein Element enthält.*

Daraus folgt unmittelbar folgender Sachverhalt. Man erinnere sich an die Notation $R^{\smile}$ für Relationen.

Proposition 4.4 *Sei $f : M \to N$ eine Funktion. Genau dann ist auch $f^{\smile} : N \to M$ eine Funktion, wenn f bijektiv ist.*

Man schreibt in dem Fall, wo $f^{\smile}$ eine Funktion ist, auch f^{-1} anstelle von $f^{\smile}$, obwohl diese nicht identisch sind. Denn $f^{\smile} : M \to N$ und $f^{-1} : \wp(N) \to \wp(M)$. (Sei zum Beispiel $f : \mathbb{Z} \to \mathbb{Z}$ die Nachfolgerfunktion. Dann ist $f^{-1}(\{1\}) = \{0\}$; aber $f^{\smile}(1) = 0$.)

Ein nicht unwichtiger Zusammenhang zwischen Surjektionen und Partitionen ist der Folgende.

Proposition 4.5 *Es sei $f : M \to N$ eine surjektive Funktion. Dann ist das System $\{f^{-1}(\{y\}) : y \in N\}$ eine Partition von M. Die dazugehörende Äquivalenzrelation auf M ist $x \sim_f y :\Leftrightarrow f(x) = f(y)$.*

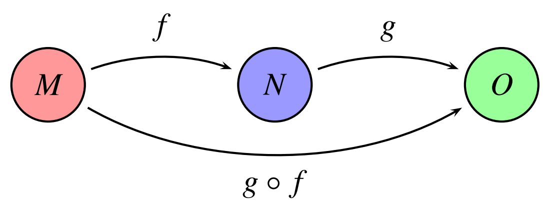
Anschaulich ist $P_y := f^{-1}(\{y\})$ ja die Menge aller $x \in M$, deren Bild unter f y ist. Keine dieser Mengen ist leer, weil f surjektiv ist. Ferner sind diese Mengen disjunkt, wenn sie verschieden sind. Denn sei $P_y \cap P_z \neq \varnothing$. Dann existiert ein $x \in M$ mit $x \in P_y$ und $x \in P_z$. Daraus folgt $f(x) = y$ und $f(x) = z$, also $y = z$ und so $P_y = P_z$.

Dazu ein Beispiel. Es ordne f jedem Indiduum seinen Geburtstag zu. Dann ist P_y die Menge aller Menschen, die denselben Geburtstag y haben. Dazu gehört die Äquivalenzrelation „hat denselben Geburtstag wie". Um Proposition 4.5 anwenden zu können, brauchen wir streng genommen, dass f surjektiv ist. Ist das nicht gegeben, so kann wie folgt verfahren: man bildet die Partitionsmengen wie oben gegeben und nimmt die leere Menge aus.

Proposition 4.6 *Es sei $f : M \to N$ eine beliebige Funktion. Dann ist das System $\{f^{-1}(\{y\}) : y \in N\} - \{\varnothing\}$ eine Partition von M. Die dazugehörende Äquivalenzrelation auf M ist $x \sim_f y :\Leftrightarrow f(x) = f(y)$.*

Im letzten Kapitel hatte ich diesen Sachverhalt schon angesprochen. Um von einer Funktion $f : M \to N$ zu zeigen, dass sie bijektiv ist (und dass somit M und N die gleiche Anzahl Elemente enthalten), reicht es, eine Abbildung $g : N \to M$ anzugeben und zu zeigen, dass $g = f^{\smile}$.

Abbildung 4.1: Funktionskomposition



Ist $f : M \to N$ und $g : N \to O$, so ist $g \circ f : M \to O$, definiert durch

(4.9) $\qquad (g \circ f)(x) := g(f(x))$

ebenfalls eine Funktion. Dies ist die sogenannte **Komposition** von f und g (Abbildung 4.1). Um zu sehen, dass dies eine Funktion ist, nehme man ein $x \in M$. Dann ist f auf x definiert und liefert den Wert $f(x)$; dieser ist eindeutig bestimmt. Auf $f(x)$ ist g definiert und liefert den Wert $g(f(x))$; dieser ist ebenfalls eindeutig bestimmt. Und so existiert $g(f(x))$ und ist eindeutig bestimmt. Ist zum Beispiel f die Funktion, die jedem Einwohner seine Adresse zuordnet, g die Funktion von Adressen zu Wahlbezirken, so ordnet $g \circ f$ jedem Einwohner seinen Wahlbezirk zu.

Ich muss an dieser Stelle eine Warnung anbringen. Funktionen sind spezielle Relationen. Insofern ist $g \circ f$ bereits definiert, wenn wir die Funktionen als Relationen sehen. In diesem Fall aber ist das Ergebnis ein anderes; denn als Relationenprodukt müssten wir es mit $f \circ g$ notieren. Jedoch gibt es eine Schwierigkeit: die Notation der Funktionen ist im Weg. Man hätte gerne – weil es so übersichtlicher ist –, dass $(g \circ f)(x) = g(f(x))$. Manche gehen deswegen den Weg, dass sie $x\,f$ schreiben anstelle von $f(x)$. Dies ist jedoch ein ziemlich ungewöhnlicher Schritt, den ich nicht gehen will. Ich behalte die übliche Schreibweise. Dies hat nur diese eine Konsequenz: die Komposition der *Funktionen* $f : M \to N$ und $g : N \to P$ wird mit $g \circ f$ bezeichnet, das Produkt der *Relationen* f und g hingegen mit $g \circ f$. (Auch dies sollte ich sagen: in beiden Fällen wird das Symbol $\circ$ verwendet, das ist so üblich.)

In dem Fall, wo $M = N = P$ können wir nicht nur die Funktion $g \circ f$ bilden sondern auch die Funktion $f \circ g$. Sie zum Beispiel $f : \mathbb{N} \to \mathbb{N} : x \mapsto x^2$ und

$g : \mathbb{N} \to \mathbb{N} : x \mapsto x + 1$. Dann ist

$$(4.10) \qquad g(f(3)) = g(9) = 10, \qquad f(g(3)) = f(4) = 16$$

Abstrakt haben wir

$$(4.11) \qquad g(f(x)) = g(x^2) = x^2 + 1, \qquad f(g(x)) = f(x + 1) = (x + 1)^2$$

Diese sind, wie das Beispiel (4.10) zeigt, verschieden. Ferner existieren auch die Funktionen $f \circ f$ und $g \circ g$. Für diese haben wir

$$(4.12) \qquad f(f(x)) = f(x^2) = (x^2)^2 = x^4, \qquad g(g(x)) = g(x+1) = (x+1)+1 = x+2$$

Noch einmal komme ich auf bijektive Funktionen zurück. Hier ist eine weitere Charakterisierung. Zunächst einmal vereinbaren wir, dass für jede Menge M die Funktion $1_M : M \to M$ die sogenannte **Identität** sein soll. Für sie gilt $1_M(x) = x$.

Proposition 4.7 *Eine Funktion $f : M \to N$ ist genau dann bijektiv, wenn es eine Funktion $g : N \to M$ gibt mit $f \circ g = 1_N$ und $g \circ f = 1_M$.*

Zum Beweis sei zunächst einmal f bijektiv. Dann setze $g := f^{\smile}$. Damit ist $g : N \to M$, und wir haben $(f \circ f^{\smile})(x) = x$ für alle $x \in N$ wie auch $(f^{\smile} \circ f)(y) = y$ für all $y \in M$. Also ist $f \circ f^{\smile} = 1_N$ und $f^{\smile} \circ f = 1_M$, wie verlangt. Jetzt sei umgekehrt g derart, dass $f \circ g = 1_N$ und $g \circ f = 1_M$. Zu zeigen ist, dass dann f bijektiv ist. Ich zeige zuerst, dass f injektiv ist. Dazu seien $x, y \in M$ mit $f(x) = f(y)$. Dann ist $x = (g \circ f)(x) = g(f(x)) = g(f(y)) = (g \circ f)(y) = y$. Also ist f injektiv. Nun gilt es noch zu zeigen, dass f surjektiv ist. Sei dazu $y \in N$. Dann ist $f(g(y)) = (f \circ g)(y) = y$. Dies bedeutet aber nichts anderes, als dass $g(y)$ ein Urbild von y ist. Dies beendet den Beweis.

Ich gebe ein instruktives Beispiel. Es sei M eine Menge und $A \subseteq M$. Setze

$$(4.13) \qquad \chi_A(x) := \begin{cases} 1 & \text{falls } x \in A \\ 0 & \text{sonst.} \end{cases}$$

Man nennt $\chi_A : M \to \{0, 1\}$ die **charakteristische Funktion** von A. Sei umgekehrt $f : M \to \{0, 1\}$ irgendeine Funktion. Dann setze

$$(4.14) \qquad \mathrm{Tr}(f) := \{x : f(x) \neq 0\}$$

Man nennt dies den **Träger** von f. Da $f(x) \neq 0$ gleichbedeutend ist mit $f(x) = 1$, haben wir

$$(4.15) \qquad \mathrm{Tr}(f) = \{x : f(x) = 1\}$$

Satz 4.8 *Die Abbildung χ, welche jeder Teilmenge A von M die charakteristische Funktion $\chi_A : M \to \{0,1\}$ zuordnet, ist bijektiv. Ihre Umkehrung ist die Abbildung* Tr.

Beweis. Sei $A \subseteq M$. Zu zeigen ist, dass $\mathrm{Tr}(\chi_A) = A$. Es ist genau dann $x \in \mathrm{Tr}(\chi_A)$, wenn $\chi_A(x) = 1$. Dies ist wiederum genau dann der Fall, wenn $x \in A$. Also ist $\mathrm{Tr}(\chi_A) = A$. Sei umgekehrt $f : M \to \{0,1\}$. Zu zeigen ist $\chi_{\mathrm{Tr}(f)} = f$. Es ist $\chi_{\mathrm{Tr}(f)}(x) = 1$ genau dann, wenn $x \in \mathrm{Tr}(f)$. Dies wiederum ist der Fall genau dann, wenn $f(x) = 1$. Also ist $\chi_{\mathrm{Tr}(f))} = f$. ⊣

Es bleibt noch eine wichtige Frage zu klären, nämlich, was wir mit Funktionen von mehreren Variablen machen. Sei zum Beispiel die folgende Funktion auf den natürlichen Zahlen gegeben.

(4.16) $$f(x,y) := x^2 + 3xy + x + 1$$

Diese ordnet ja jetzt *zwei* natürliche Zahlen einen Wert zu, nicht nur einer. Jedoch können wir diese beiden in ein Paar zusammenfassen, und dann ist die Funktion eine Zuordnung der natürlichen Zahl $x^3 + 3xy + x + 1$ zu dem Paar $\langle x, y\rangle$. Wir schreiben also

(4.17) $$f : \mathbb{N} \times \mathbb{N} \to \mathbb{N} : \langle x, y\rangle \mapsto x^2 + 3xy + x + 1$$

Als Menge sieht diese dann so aus:

(4.18) $$\{\langle\langle 0,0\rangle, 1\rangle, \langle\langle 1,0\rangle, 3\rangle, \langle\langle 0,1\rangle, 1\rangle, \langle\langle 2,0\rangle, 7\rangle, \langle\langle 1,1\rangle, 6\rangle, \cdots\}$$

Falls wir eines der Argumente festhalten, also etwa $x := 1$ setzen, so bekommen wir eine einstellige Funktion, die wir $f(1,y)$ schreiben. Diese hat die Vorschrift $y \mapsto 3y + 3$ (indem wir in $x^2 + 3xy + x + 1$ für x 1 einsetzen), und sieht als Menge so aus:

(4.19) $$\{\langle 0,3\rangle, \langle 1,6\rangle, \langle 2,9\rangle, \cdots\}$$

Ebenso können wir y festlegen, etwa auch $y := 1$ setzen. Dann bekommen wir die Funktion $f(x,1) : x \mapsto x^2 + 4x + 1$, also

(4.20) $$\{\langle 0,1\rangle, \langle 1,6\rangle, \langle 2,13\rangle, \cdots\}$$

Wie wir sehen, ist der oben definierte Funktionsbegriff tatsächlich allgemein genug!

Ich schließe mit einem Exkurs über definite Kennzeichnungen. In der natürlichen Sprache gebrauchen wir gerne Sprechweisen wie „der Schwager des besten Freundes meines Nachbarn”. Diese suggerieren, dass es ein Objekt gibt, welches durch die Kennzeichnung eindeutig bestimmt ist. Dies ist nur dann der Fall, wenn die Beschreibung ein und nur ein Objekt zum Gegenstand hat wie etwa „mein Geburtstag”. Wenn ich von meinem Geburtstag spreche, so ist dies ein Tag im Jahr, der nicht nur existiert sondern auch eindeutig ist. Eine definite Kennzeichnung kann auf zwei Weisen verunglücken. Die erste ist, wenn es gar kein Objekt gibt, auf das die Eigenschaft verweist (etwa „mein Auto”, denn ich habe keins). Russell benutzte hier das Beispiel vom „heutigen König von Frankreich”, der nicht existiert. Die zweite Möglichkeit ist, wenn es mehr als ein Objekt gibt. Etwa in dem Ausdruck „mein Nachbar”, denn ich habe mehrere Nachbarn. In der Theorie bezeichnet ein Ausdruck „der Φ” oder „mein Φ” nichts, wenn Φ nicht auf genau ein Objekt zutrifft. In der Mathematik gibt es haufenweise solcher definiter Kennzeichnungen. Eine davon ist versteckt in „$f(x) = y$”. Sofern f eine Relation ist, bezeichnet $f(x)$ dasjenige y mit $\langle x, y\rangle \in f$. Damit das für alle $x \in M$ wohldefiniert ist, muss f eine Funktion auf M sein.

Übungen

[14] Es sei $M = \{\clubsuit, \spadesuit, \heartsuit, \diamondsuit\}$, $N = \{0, 1\}$.

1. Ist $f = \{\langle\clubsuit, 0\rangle, \langle\spadesuit, 0\rangle, \langle\heartsuit, 0\rangle\}$ eine Funktion?
2. Ist $g = \{\langle\clubsuit, 0\rangle, \langle\spadesuit, 0\rangle, \langle\heartsuit, 0\rangle, \langle\diamondsuit, 0\rangle\}$ eine Funktion?
3. Ist $h = \{\langle\clubsuit, 1\rangle, \langle\spadesuit, 0\rangle, \langle\heartsuit, 1\rangle, \langle\diamondsuit, 0\rangle\}$ eine Funktion?

Bitte geben Sie stets eine kurze Begründung.

[15] Es sei $f = \{\langle a, 0\rangle, \langle b, 0\rangle, \langle c, 1\rangle, \langle d, 1\rangle, \langle e, 1\rangle\}$ eine Funktion nach $N = \{0, 1\}$. Was ist der Definitionsbereich von f? Bestimmen Sie die von f auf M induzierte Partition. Wie kann man anhand der Partition entscheiden, ob f injektiv ist? Gibt es noch eine andere Funktion g nach N, die dieselbe Partition erzeugt? Wenn ja, können Sie g angeben?

[16] Es seien f und g Funktionen von der Menge der rationalen Zahlen in die Menge der rationalen Zahlen. Dabei sei $f : x \mapsto 2x - 1$ und $g : x \mapsto x^2$.

1. Bestimmen Sie die Umkehrfunktion von f.

2. Zeigen Sie, dass $g^{\smile}$ keine Funktion ist.

[17] Für die Funktionen f und g aus der vorigen Aufgabe: Bestimmen Sie $f \circ g$, $f \circ f$, $g \circ g$ und $g \circ f$.

Kapitel 5

Rekursion und Induktion

Rekursion spielt nicht nur in der Mengenlehre eine wichtige Rolle. Aus der Mathematik und Informatik ist sie gar nicht wegzudenken. Die **Rekursion** ist ein Definitionsschema, das einen Gegenstand nicht direkt definiert sondern mit Hilfe anderer Gegenstände, die ihrerseits oft nicht direkt definiert sind. Genauer ist ein Rekursionsschema eine Definition, bei der das Definiendum (das zu definierende) wieder im Definiens (der definierende Ausdruck) auftaucht, wobei aber jetzt ein Parameter (in der Regel eine Zahl) abgenommen hat. Solange die Parameter letztendlich „auf dem Boden ankommen" und für diesen Fall eine explizite Definition bereitsteht, ist eine solche Definition völlig eindeutig und risikolos. Als Beispiel möge hier die Definition der *Exponentiation* dienen.

$$\begin{aligned} n^0 &:= 1 \\ n^{m+1} &:= n \cdot n^m \end{aligned} \tag{5.1}$$

Noch einmal anders aufgeschrieben sieht das so aus.

> Das Definiendum ist der Ausdruck „n^m". Dieser hat zwei Parameter, n und m. Das Definiens sagt: Ist der Parameter m gleich 0, so gebe ich den Wert direkt an; dieser ist 1. Ist der Parameter > 0, so gebe ich den Wert unter der Benutzung des Wertes von n^{m-1} an. Dann nämlich sei n^m gleich n multipliziert mit n^{m-1}.

Der Witz an der Gleichung 5.1 ist der, dass nur der Ausdruck n^0 direkt definiert ist: Er ist gleich 1 (übrigens auch, wenn $n = 0$). Der Ausdruck n^m kann für $m > 0$ nur so berechnet werden, dass man zunächst $n^{(m-1)}$ ausrechnet und dann mit n multipliziert. (In der Gleichung oben habe ich anstelle von n^m n^{m+1} gewählt, weil das garantiert, dass im Exponenten eine Zahl > 0 steht.) Das sieht zirkulär aus; augenscheinlich definiert man die Exponentiation mittels der Exponentiation selbst. Aber trotzdem kann man jede Exponentiation exakt lösen. Denn die Exponentiation mit der Zahl m wird ja durch die Exponentiation mit der Vorgängerzahl $m - 1$ erklärt. Und wenn $m > 0$ eine natürliche Zahl ist, so kommt auf diesem Wege tatsächlich zu einem expliziten Ergebnis. Man steigt einfach die Treppe abwärts: von m zu $m - 1$, dann zu $m - 2$, und so weiter, bis man bei 0 ankommt. Und dann bekommt man einen Wert, den man schließlich einsetzen kann. Hier ist ein Beispiel.

$$
\begin{aligned}
7^4 &= 7 \cdot 7^3 \\
&= 7 \cdot 7 \cdot 7^2 \\
&= 7 \cdot 7 \cdot 7 \cdot 7^1 \\
&= 7 \cdot 7 \cdot 7 \cdot 7 \cdot 7^0 \\
&= 7 \cdot 7 \cdot 7 \cdot 7 \cdot 1 \\
&= 2401
\end{aligned}
\tag{5.2}
$$

Definition 5.1 (Rekursion) *Eine Definition einer Funktion f auf natürlichen Zahlen in irgendeine Menge ist* ***rekursiv****, falls der Wert $f(n)$ unter Zuhilfenahme der Zahl n sowie der Werte von $f(m)$ für $m < n$ definiert wird. Dabei muss $f(n)$ eindeutig bestimmt sein, falls nur die Werte n, $f(0)$, $f(1)$, $\cdots$, $f(n-1)$ gegeben sind.*

Dies sollte ich noch einmal erläutern. Zunächst einmal ist nicht eine Funktion an und für sich rekursiv sondern nur die Definition dieser Funktion. Die Definition wird mit Hilfe von Rekursionsgleichungen gegeben. Ein Beispiel dafür ist das Gleichungssystem 5.1. Dort steht f stets links von :=. Gegeben ein Zahl k, kann man nur *eine* der Gleichungen anwenden, um den Wert von $f(k)$ auszurechnen. Auf der rechten Seite darf auch f stehen, aber es darf nur auf Werte $< k$ angewendet. Wenn dies gegeben ist, so existiert eine Lösung für das System. Ein und dieselbe Funktion kann eine rekursive Definition besitzen und dennoch direkt, das heißt ohne Verwendung der Funktionswerte $f(m)$, berechnet werden. Beispiele, die dies illustrieren, werden noch folgen. Zweitens gibt es einen Sonderfall zu betrachten. Aus der Definition folgt, dass $f(0)$ direkt gegeben sein muss. Denn es gibt keine natürliche Zahl $m < 0$. Das heißt, $f(0)$ muss auch in einer rekursiven Definition direkt gegeben sein. Ich weise auf folgendes Faktum hin.

[Eindeutigkeit der Lösung]

> Es sei f rekursiv durch ein System von Rekursionsgleichungen definiert. Dann gilt für für jede Funktion g, die dieses System von Rekursionsgleichungen erfüllt, dass $g = f$, das heißt, für jede natürliche Zahl k ist $g(k) = f(k)$.

Ich weise der Vollständigkeit halber darauf hin, dass wir hier nur sogenannte primitive Rekursion behandeln; das ist der Fall, wo wir zur Berechnung von $f(n)$ lediglich den Wert von $f(n-1)$ benötigen, sofern $n > 0$. Ein berühmtes Beispiel einer nicht primitiven Rekursion ist die Fibonacci-Folge.

$$\begin{aligned} F(0) &:= 0 \\ F(1) &:= 1 \\ F(n+2) &:= F(n+1) + F(n) \end{aligned} \tag{5.3}$$

Wie man leicht nachrechnet, bestimmt dies die folgene Funktion (hier als Folge ihrer Funktionswerte notiert):

$$0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, \ldots \tag{5.4}$$

Es gibt sehr viele rekursive Definitionen, die auf dem ersten Blick gar nicht so aussehen. Ich veranschauliche das am Beispiel, den sogenannten Dreieckszahlen. Anschaulich ist die $n$te Dreieckszahl, $D(n)$, die Summe $1 + 2 + 3 + \cdots + n$. Diese Definition mag uns befriedigen, aber Mathematiker (und Computer) sehen in den Punkten lediglich eine bequeme Schreibweise, die nicht wirklich definiert ist. Man verlässt sich da des öfteren auf die Intuition. Hier ist dagegen eine Definition, die vollends explizit ist und nichts dem Zufall überlässt.

$$\begin{aligned} D(0) &:= 0 \\ D(n+1) &:= D(n) + n + 1 \end{aligned} \tag{5.5}$$

Dies ist rekursiv: Was $D(n+1)$ ist, wird mit Hilfe von $D(n)$ erklärt. Die Zahlen heißen übrigens Dreieckszahlen, weil $D(n)$ gerade die Anzahl von Kugeln in einem Dreieck der Größe n angibt.

$$\begin{array}{ll} n = 0 & \\ n = 1 & \bullet \\ n = 2 & \bullet\ \bullet \\ n = 3 & \bullet\ \bullet\ \bullet \\ n = 4 & \bullet\ \bullet\ \bullet\ \bullet \\ n = 5 & \bullet\ \bullet\ \bullet\ \bullet\ \bullet \\ n = 6 & \bullet\ \bullet\ \bullet\ \bullet\ \bullet\ \bullet \\ n = 7 & \bullet\ \bullet\ \bullet\ \bullet\ \bullet\ \bullet\ \bullet \end{array} \tag{5.6}$$

Ich zeige nun den folgenden Sachverhalt:

Proposition 5.2 $D(n) = \frac{n(n+1)}{2}$.

Zum Beweis zeige ich, dass die gegebene Funktion $g(n) := \frac{n(n+1)}{2}$ die Rekursionsgleichungen 5.5 erfüllt, wobei wir anstelle vom Symbol D jetzt natürlich g schreiben müssen und „=” anstelle von „:=”, da g ja bereits definiert ist:

$$\begin{aligned} g(0) &= 0 \\ g(n+1) &= g(n) + n + 1 \end{aligned} \tag{5.7}$$

Nun setzen wir die explizite Definition ein:

$$\begin{aligned} \tfrac{0(0+1)}{2} &= 0 \\ \tfrac{(n+1)(n+1+1)}{2} &= \tfrac{n(n+1)}{2} + n + 1 \end{aligned} \tag{5.8}$$

Die erste Gleichung ist sicherlich richtig. Die zweite forme ich noch ein wenig um, indem ich anstelle von $n + 1 + 1$ den Term $n + 2$ setze. Es ist also noch dies zu zeigen.

$$\frac{(n+1)(n+2)}{2} = \frac{n(n+1)}{2} + n + 1 \tag{5.9}$$

Jetzt rechnen wir ein wenig.

$$\frac{(n+1)(n+2)}{2} = \frac{(n+2)(n+1)}{2} = \frac{n(n+1)}{2} + \frac{2(n+1)}{2} = \frac{n(n+1)}{2} + n + 1 \tag{5.10}$$

Und dies ist offensichtlich, was wir zeigen sollten.

Wenn man Zahlen rekursiv definiert, lassen sich Eigenschaften darüber oft nicht direkt zeigen. Oft greift man deswegen zu einem *Beweis durch Induktion*.

> [Induktionsprinzip]
>
> Eine Behauptung A gilt genau dann für alle natürlichen Zahlen, wenn (1) $A(0)$ der Fall ist und (2) für alle natürlichen Zahlen n gilt: Ist $A(n)$ der Fall, so auch $A(n + 1)$.

(Dies ist kein Satz, weil wir diese Behauptung nicht beweisen können. Sie ist nämlich konstitutiv für die natürlichen Zahlen, das heißt, wir charakterisieren die natürlichen Zahlen dadurch, dass dieses Prinzip gikt. Es ist bekannt als das Postulat von Peano, nach Giuseppe Peano (1858 - 1932).) Das Induktionsprinzip ist verwandt mit der Rekursion, aber nicht identisch. Der Schritt (1) heißt **Induktionsanfang**, der Schritt (2) **Induktionsschritt**. Die Bedingung „falls $A(n)$ der Fall ist" ist der eigentliche Witz; oft kann man $A(n+1)$ gar nicht direkt zeigen. Aber es genügt eben, wenn man sie zeigen kann, wenn nur $A(n)$ gilt (wobei man Letzteres eben nicht zeigen muss, sondern in diesem Fall einfach annimmt). Ist der Beweis fertig, hat man im Ergebnis Folgendes. $A(0)$ gilt, weil es direkt gezeigt wurde. $A(1)$ gilt, weil $A(0)$ gilt. (Man setze dazu $n = 0$ in (2).) $A(2)$ gilt, weil $A(1)$ gilt. (Man setze dazu $n = 1$ in (2).) $A(3)$ gilt, weil $A(2)$ gilt. Und so weiter. Man kann das Induktionsprinzip verwenden um zu zeigen, dass Rekursionsgleichungen nur eine Lösung haben.

Ich zeige dies an einem einfachen Beispiel, dem Satz 2.6. Dieser Satz sagt, dass $\wp(M)$ 2^n Elemente hat, wenn M n Elemente hat. Dies beweise ich induktiv. Wir beginnen also mit dem Fall $n = 0$. Hier ist $M = \varnothing$, denn 0 Elemente zu haben, heißt, leer zu sein. Es ist $\wp(\varnothing) = \{\varnothing\}$. Diese Menge hat 1 Element. Gleichzeitig ist $2^0 = 1$. Dies ist der Induktionsanfang. Nun zum Induktionsschritt. Es sei nun bereits gezeigt, dass $\wp(S)$ 2^n Elemente hat, wenn S n Elemente hat. Nun habe M $n + 1$ Elemente. Ich wähle eines aus, etwa m, und zerlege M als $M = S \cup \{m\}$, wo S das Element m nicht enthält (dh $S = M - \{m\}$). Es hat S n Elemente, daher hat $\wp(S)$ nach Induktionsvoraussetzung 2^n Elemente. Nun schauen wir uns $\wp(M)$ an. Sei T eine Teilmenge von M. (Fall 1) T enthält *m nicht*. Dann ist T Teilmenge von S. Es gibt 2^n solcher Mengen. (Fall 2) T enthält m. Dann setze $T^- := T - \{m\}$. T^- ist eine Teilmenge von S. Ferner ist $T = T^- \cup \{m\}$. Die Abbildung $T \mapsto T^-$ ist somit bijektiv, da sie eine Umkehrabbildung besitzt. Hier noch einmal ausführlich die Begründung. Sei U eine Teilmenge von M, die m enthält und sei $U \neq T$. Dann muss ein Element g existieren, das entweder (a) in U ist aber nicht in T, oder (b) in T ist aber nicht in U. Gewiss ist $g \neq m$. Dann ist im Fall (a) $g \in U^-$ aber $g \notin T^-$ und im Fall (b) $g \in T^-$ aber $g \notin U^-$. Also ist $U^- \neq T^-$. Umgekehrt ist für jede Teilmenge V von S die Menge $V^+ := V \cup \{m\}$ eine Teilmenge von M, die m enthält. Wiederum ist mit $V \neq W$ auch $V \cup \{m\} \neq W \cup \{m\}$. Es gibt also 2^n Teilmengen von M, die m enthalten. Daraus folgt unmittelbar, dass es $2^n + 2^n = 2 \cdot 2^n = 2^{n+1}$ Teilmengen von M gibt. Die letzte Gleichung ist unmittelbar die Rekursionsgleichung aus (5.1).

Noch ein Beispiel. Ich behaupte: *Ist n ungerade, so ist auch n^m ungerade für jedes m.* Zum Beweis setzen wir erst $m = 0$. In diesem Fall ist $n^m = n^0 = 1$, eine ungerade Zahl. Nun sei $m > 0$ beliebig. Wir nehmen nun an, $n^{(m-1)}$ sei ungerade (das ist die Induktionsannahme). Dann ist $n^m = n \cdot n^{(m-1)}$ auch ungerade, weil es das Produkt zweier ungerader Zahlen ist.

Aber nicht nur (scheinbar) leichte Tatsachen kann man so beweisen. Etwas schwieriger zu zeigen ist zum Beispiel folgender Sachverhalt: *Ist* $n > 1$*, so ist* $n^m \geq n \cdot m$. Beginnen wir wieder mit $m = 0$. Es ist $n^m = 1 > 0 = n \cdot m$. Wir zeigen die Behauptung auch noch für $m = 1$: $n^m = n = n \cdot m$. Sei die Behauptung nun für $m - 1$ gezeigt. Wir dürfen annehmen, dass $m > 1$ ist, das heißt $m - 1 \geq 1$. Dann ist $n^m = n \cdot n^{(m-1)} \geq 2 \cdot n^{(m-1)} \geq 2 \cdot n \cdot (m-1) \geq (n \cdot (m-1)) + n = n \cdot m$. Hierbei ist die erste Gleichung einfach die Definition; die erste Ungleichung gilt, weil $n \geq 2$ vorausgesetzt war. Die zweite Ungleichung gilt nach Induktionsannahme. Die dritte Ungleichung benutzt $n \cdot (m-1) \geq n \cdot 1 = n$.

Ein dritter Fall betrifft die Dreieckszahlen. Ich zeige

$$D(n) = \frac{n(n+1)}{2} \tag{5.11}$$

Dies gilt für $n = 0$. Denn dann steht da $D(0) = 0(0+1)/2 = 0$. Nun sei die Behauptung für $n + 1$ gezeigt unter der Bedingung, dass sie für n gilt.

$$\begin{aligned} D(n+1) = D(n) + n + 1 = \frac{n(n+1)}{2} + (n+1) &= \left(\frac{n}{2} + 1\right)(n+1) \\ &= \left(\frac{n+2}{2}\right)(n+1) = \frac{(n+1)(n+2)}{2} \end{aligned} \tag{5.12}$$

Die erste Gleichheit gilt nach Definition von $D(n+1)$, die zweite gilt nach der Annahme über $D(n)$, die wir gemacht haben. Der Rest ist reine Umformung.

Ich stelle hier noch eine Notation vor, die in der Wissenschaft sehr üblich ist, und die man unbedingt kennen sollte. Es handelt sich um die Summe „$\sum_{i=0}^{k} a_i$". Man stellt dies normalerweise so dar: Wir haben eine Folge von Zahlen a_0, a_1, a_2, $\cdots$ (notiert $(a_i)_{i \in \mathbb{N}}$) und bilden dann die sogenannte **Summenfolge**.

$$\sum_{i=0}^{k} a_i := a_0 + a_1 + \cdots + a_k \tag{5.13}$$

Wie schon betont, sind die drei Punkte für Mathematiker unbefriedigend. Anstelle dessen definieren wir eine Funktion $s(k)$ auf folgende Weise.

$$\begin{aligned} s(0) &:= a_0 \\ s(k+1) &:= s(k) + a_{k+1} \end{aligned} \tag{5.14}$$

Dann gilt

$$\sum_{i=0}^{k} a_i = s(k) \tag{5.15}$$

Dies können wir nicht wirklich beweisen, weil die rekursive Definition (5.14) die intuitive Version (5.13) ja ersetzt. Man beachte, dass die Folge der a_i dabei vorgegeben ist. Alternativ können wir auch anstelle von einer Folge $(a_i)_{i\in\mathbb{N}}$ auch von einer Funktion g ausgehen. Dann sieht die Definition 5.14 so aus:

$$\begin{aligned} s(0) &:= g(0) \\ s(k+1) &:= s(k) + g(k+1) \end{aligned} \tag{5.16}$$

Wie gesagt benutzt man die Schreibweise „$\sum_{i=0}^{k} g(i)$" anstelle von $s(k)$. Die Dreieckszahlen kann man also auch wie folgt beschreiben:

$$D(n) = \sum_{i=0}^{n} i \tag{5.17}$$

Um die Wichtigkeit der Rekursion zu verstehen, sollte man wissen, dass sich alleine aus der Nachfolgerfunktion Addition und Multiplikation definieren lassen. Es sei zu einer Zahl n' der Nachfolger, also ist stets $n' = n + 1$. Dann *definieren* wir rekursiv die Addition mittels der Nachfolgerfunktion wie folgt.

$$\begin{aligned} m + 0 &:= m \\ m + n' &:= (m+n)' \end{aligned} \tag{5.18}$$

Das sieht sehr geheimnisvoll aus. Zunächst einmal muss man sich bei dieser Definition dumm stellen: wir tun so, als sei die Addition nicht vorhanden, nur die Nachfolgerfunktion. Jetzt sagen wir, was es heißt, zwei Zahlen m und n zu addieren. Der erste Fall ist $n = 0$. In diesem Fall ist die Summe genau n. Ist dann m nicht 0, so ist n der Nachfolger einer Zahl k, also $n = k'$. Nehmen wir nun an, wir wüssten schon, was $m + k$ ist. Dann sagt uns die zweite Zeile, was $m + n$ ist. Dazu sollen wir nach dieser Vorschrift den Nachfolger von $m + k$ nehmen.

$$
\begin{aligned}
3 + 4 &= 3 + 3' \\
&= (3 + 3)' \\
&= (3 + 2')' \\
&= ((3 + 2)')' \\
&= (((3 + 1')')' \\
&= (((3 + 1)')')' \\
&= (((3 + 0')')')' \\
&= (((3 + 0)')')')' \\
&= 3'''' \\
&= 4''' \\
&= 5'' \\
&= 6' \\
&= 7
\end{aligned} \tag{5.19}
$$

Eine etwas effizientere Definition (selbstverständlich mit gleichem Ergebnis) ist wie folgt.

$$
\begin{aligned}
m + 0 &:= m \\
m + n' &:= m' + n
\end{aligned} \tag{5.20}
$$

Hier sieht die Rechnung wie folgt aus.

$$
\begin{aligned}
3 + 4 &= 3 + 3' \\
&= 3' + 3 \\
&= 4 + 3 \\
&= 4 + 2' \\
&= 4' + 2 \\
&= 5 + 2 \\
&= 5 + 1' \\
&= 5' + 1 \\
&= 6 + 1 \\
&= 6 + 0' \\
&= 6' + 0 \\
&= 7 + 0 \\
&= 7
\end{aligned} \tag{5.21}
$$

Hat man Addition definiert, kann man die Multiplikation mit Hilfe der Addition definieren.

$$
\begin{aligned}
m \cdot 0 &:= 0 \\
m \cdot n' &:= (m \cdot n) + m
\end{aligned} \tag{5.22}
$$

Am Anfang des Kapitels hatte ich die Exponentiation definiert. Wer schwindelfrei ist, mag sich auch noch folgendes Beispiel ansehen, den sogenannten **Exponentialturm**. Er wird definiert durch

$$\begin{aligned} t(m,0) &:= 1 \\ t(m,n+1) &:= m^{t(m,n)} \end{aligned} \tag{5.23}$$

Daraus ergibt sich $t(2,1) = 2^{t(2,0)} = 2^1 = 2$, $t(2,2) = 2^{t(2,1)} = 2^2 = 4$, $t(2,3) = 2^{t(2,2)} = 2^4 = 16$, $t(2,4) = 2^{t(2,3)} = 2^{16} = 65536$. Und $t(2,5)$ ist ... 2^{65536}, eine riesige Zahl (größer als 10^{20000})!

Wir sehen also, dass man aus der Nachfolgerfunktion eigentlich alles aufbauen kann. Deswegen nimmt sie eine zentrale Stelle ein. Das Großartige ist nun, dass wir damit zeigen können, dass für das Rechnen nur eine einzige Eigenschaft wichtig ist: wir müssen wissen, welche Zahl am Anfang steht (wir nennen sie die „Null"), und welche Zahl auf eine gegebene Zahl folgt. Cantor nennt dies eine **Zählreihe**.

$$0, 1, 2, 3, 4, 5, \cdots \tag{5.24}$$

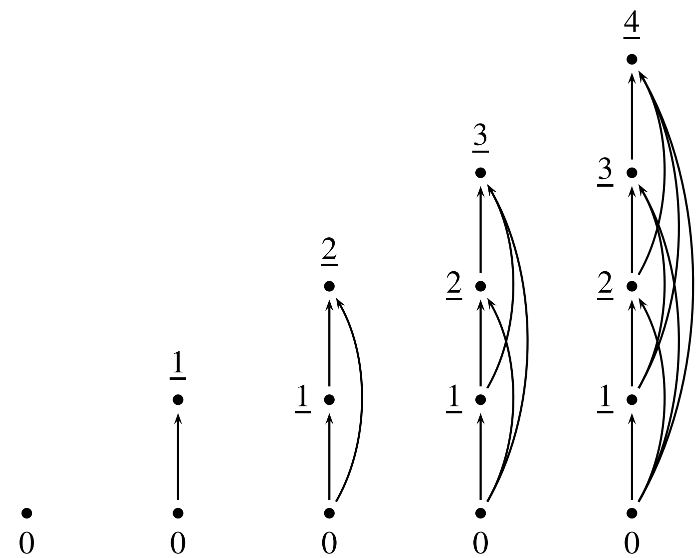
Eine Zählreihe ist eine unendlich aufsteigende Folge, die ein kleinstes aber kein größtes Element hat. Das wichtigste Beispiel sind die natürlichen Zahlen. Die Namen der Zahlen sind dabei unerheblich, nur ihre Ordnung ist relevant. Sie bestimmt, wie die Nachfolgerfunktion aussieht. Hat man diese bestimmt, ist sämtliches Rechnen bereits vollständig bestimmt: Addition, Multiplikation, Exponentiation. Die praktische Ausführung ist dabei etwas anderes; wir lernen, Zahlen in Zehnerschreibweise zu benennen und mit ihnen so zu rechnen.

Die Rekursion lässt sich nun nicht alleine dafür nehmen, um Zahlen zu definieren. Sie erlaubt, Folgen von Gegenständen gleich welcher Art zu definieren. Ich gebe ein Beispiel. In der Mengenlehre kann man die Rekursion benutzen, um spezielle Mengen zu definieren. Ein Beispiel ist die Definition der natürlichen Zahlen als Mengen:

$$\begin{aligned} \underline{0} &:= \varnothing \\ \underline{n+1} &:= \underline{n} \cup \{\underline{n}\} \end{aligned} \tag{5.25}$$

Wir nennen die so definierten Zahlen **Ordinalzahlen**. (Man nennt sie auch **Kardinalzahlen**; die Begriffe sind verschieden, für endliche Mengen allerdings gleich.) Dies ist also einerseits eine Definition von gewissen Mengen. Andererseits steht jetzt auch da, wie der Nachfolger von $\underline{n}$ aussieht: es ist $\underline{n+1} = \underline{n} \cup \{\underline{n}\}$. Und damit

Abbildung 5.1: Kardinalzahlen als Graphen



stehen uns jetzt innerhalb der Mengen (mit rekursiver Definition) die Addition, Multiplikation, Exponentiation und so weiter!

Daraus folgt zum Beispiel

(5.26) $\underline{1} = \underline{0} \cup \{\underline{0}\} = \varnothing \cup \{\varnothing\} = \{\varnothing\}$

Und weiter ist

(5.27) $\underline{2} = \underline{1} \cup \{\underline{1}\} = \{\varnothing\} \cup \{\{\varnothing\}\} = \{\varnothing, \{\varnothing\}\}$

Es ist vielleicht ganz instruktiv, diese Kardinalzahlen als Graphen aufzuzeichnen, siehe Abbildung 5.1. Diese Definition ist ziemlich ausgeklügelt. Die Menge $\underline{n}$ hat zum Beispiel genau n Elemente. Für $n = 0$ trifft das sicher zu. Nun sei die Behauptung für $\underline{n}$ gezeigt. Es ist $\underline{n+1} = \underline{n} \cup \{\underline{n}\}$. Nach Induktionsannahme hat $\underline{n}$ genau n Elemente. Damit jetzt $\underline{n+1}$ genau $n+1$ Element hat, müssen wir zeigen, dass $\underline{n} \notin \underline{n}$ ist. Dies wird aber bei Mengen stets vorausgesetzt. Es ist ein eisernes Gesetz, dass eine Menge sich nicht selbst enthalten kann. Außerdem sind die Mengen $\underline{n}$ transitiv:

Proposition 5.3 *Die Mengen $\underline{n}$ sind* ***transitiv****, das heißt, ist $S \in P \in \underline{n}$, so ist auch $S \in \underline{n}$.*

Beweis. Induktion über n. Ist $n = 0$ so ist $\underline{n} = \varnothing$ und deswegen $P \in \underline{n}$ nie der Fall, also die Behauptung richtig. Nun sei die Behauptung für n gezeigt, und es sei $S \in P \in \underline{n+1}$. Nun ist $\underline{n+1} = \underline{n} \cup \{n\}$ und so ist entweder (a) $P \in \underline{n}$ oder (b) $P = \underline{n}$. Fall (a). Ist $S \in P$, so ist $S \in \underline{n}$, weil $\underline{n}$ transitiv ist. Nun ist $\underline{n} \subseteq \underline{n+1}$ (das folgt unmittelbar aus (5.25)), also $S \in \underline{n+1}$. (b) Ist $S \in P$, dann ist $S \in \underline{n}$. Wie in (a) ist dann aber schon $S \in \underline{n+1}$. ⊣

Daraus bekommen wir folgende Eigenschaft.

Proposition 5.4 *Genau dann ist $P \in \underline{n}$, wenn $P = \underline{k}$ für ein $k < n$. Kurz:*

(5.28) $$\underline{n} = \{\underline{k} : k < n\}$$

Beweis. Wiederum Beweis durch Induktion. Die Behauptung ist für $n = 0$ ist sicher richtig. Sei sie gezeigt für n; wir zeigen sie für $n+1$. Wir zeigen zuerst ($\subseteq$). Sei also $P \in \underline{n+1} = \underline{n} \cup \{\underline{n}\}$. Dann ist (a) $P \in \underline{n}$ oder (b) $P = \underline{n}$. Im Falle (a) ist wegen der Induktionsannahme $P = \underline{k}$ für ein $k < n$, also $k < n+1$. Im Falle (b) ist die Behauptung auch richtig, denn $n < n+1$. Jetzt zeigen wir noch ($\supseteq$). Es sei umgekehrt $k < n+1$. Wir müssen dann $\underline{k} \in \underline{n}$ zeigen. Fall 1. $k < n$. Dann ist $\underline{k} \in \underline{n}$ nach Induktionsannahme. Da $\underline{n} \subseteq \underline{n+1}$, ist auch $\underline{k} \in \underline{n+1}$. Fall 2. $k = n$. Da $\underline{n+1} = \underline{n} \cup \{\underline{n}\}$ ist sicher $\underline{k} \in \underline{n+1}$. Und das war zu zeigen. ⊣

Ich mache darauf aufmerksam, dass historisch gesehen eigentlich die Gleichung (5.28) am Anfang stand. Daraus lässt sich die Rekursionsvorschrift denn auch leich herleiten. Denn es ist sicher $\varnothing$ die Menge aller natürlichen Zahlen kleiner als Null. Für $\underline{n+1}$ schließlich bekommen wir Folgendes.

(5.29) $$\begin{aligned} \underline{n+1} &= \{\underline{k} : k < n+1\} \\ &= \{\underline{k} : k < n\} \cup \{\underline{n}\} \\ &= \underline{n} \cup \{\underline{n}\} \end{aligned}$$

Hier haben wir zweimal (5.28) angewandt. Daraus gewinnen wir wiederum

Proposition 5.5 *Ist $P \in \underline{n}$, so ist auch $P \subseteq \underline{n}$.*

Beweis. Aus dem vorigen Satz leiten wir ab, dass $P = \underline{k}$ für ein $k < n$. Außerdem ist $P = \{\underline{i} : i < k\}$ und $\underline{n} = \{\underline{i} : i < n\}$; und da $k < n$, ist dann $P \subseteq \underline{n}$. ⊣

Den vorigen Satz hätten wir auch anders zeigen können. Denn das folgt unmittelbar aus der Tatsache, dass $\underline{n}$ transitiv ist. Denn $P \subseteq \underline{n}$ heißt ja nicht anderes, als dass jedes Element von P auch Element von $\underline{n}$ ist.

Ich ziehe noch einen wichtigen Schluss, der zeigt, dass die Ordinalzahlen auch die Ordnung der Zählreihe in sich tragen.

Proposition 5.6 *Genau dann ist $m < n$, wenn $\underline{m} \in \underline{n}$.* ⊣

Man kann also rein intrinsisch sagen, was eine Menge für Eigenschaften haben muss, um von der Form $\underline{n}$ zu sein. Dies erlaubt schließlich, die gesamte Arithmetik als Spiel zwischen Mengen aufzufassen!

Übungen

[18] Es sei folgende Vorschrift gegeben.

$$\begin{aligned} F(0) &:= 0 \\ F(n+1) &:= F(n) + 3^n \end{aligned} \tag{5.30}$$

Berechnen Sie die ersten fünf Folgenglieder: $F(0), \cdots, F(4)$.

[19] Zeigen Sie, dass $F(n) = \frac{3^n-1}{2}$. *Anleitung.* Sei $G(n) := \frac{3^n-1}{2}$. Rechnen Sie nach, dass $G(0) = 0$; zeigen Sie zweitens, dass $G(n+1) = G(n) + 3^n$. Damit erfüllt G die Rekursionsgleichung (5.30) und ist identisch mit F.

[20] Berechnen Sie die Differenz zwischen den aufeinanderfolgenden Quadratzahlen: $5^2 - 4^2$, $4^2 - 3^2$, $3^2 - 2^2$, $2^2 - 1^2$, $1^2 - 0^2$. Stellen Sie anschließend $(n+1)^2 - n^2$ als Funktion von n dar.

[21] Mit Hilfe der vorigen Aufgabe wollen wir eine Rekursionsgleichung für n^2 aufstellen. Wir wählen ein Funktionssymbol, H, für das wir die Gleichung aufstellen, deren einzige Lösung natürlich $H(n) = n^2$ sein wird. Da $0^2 = 0$, setzen wir $H(0) = 0$. Jetzt brauchen wir eine zweite Gleichung. Nach der vorigen Aufgabe hat sie die Form: $H(n+1) = H(n)+?$ für einen gewissen Summanden, der in der vorigen Aufgabe berechnet wurde.

[22] Wir definieren analog zur Summenfolge die Produktfolge.

$$\begin{aligned} p(0) &:= a_0 \\ p(k+1) &:= p(k) \cdots a_{k+1} \end{aligned} \tag{5.31}$$

Man bezeichnet $p(k)$ auch durch „$\prod_{i=0}^{k} a_i$". Zeigen Sie, dass $m^{n+1} = \prod_{i=0}^{n} m$.

Kapitel 6

Wörter und Sprachen

In diesem und den folgenden Kapiteln werden wir uns mit *Sprachen* befassen. Sprachen sind Mengen von Zeichenketten. Zeichenketten wiederum sind Ketten von Zeichen. Womit wir bei dem Begriff der *Kette* sind. Eine (endliche) Kette ist dadurch ausgezeichnet, dass sie einen Anfang und ein Ende hat. (Ich ignoriere hier eine andere mögliche Struktur, zum Beispiel die einer Halskette, wo es eigentlich keinen ausgezeichneten Anfangs- und Endpunkt gibt. Diese denken wir uns — wie im wirklichen Leben — aus der hier besprochenen Kette dadurch hergestellt, dass Anfang und Ende nachträglich verbunden werden. Ferner werden wir hier nur endliche Ketten betrachten.)

In einer Kette hat jedes Kettenglied, das nicht am Anfang und nicht am Ende ist, einen linken und einen rechten Nachbarn. Texte können wir auch als Ketten begreifen. Sie beginnen beim ersten Buchstaben und ziehen sich bis zum letzten durch. Im Deutschen ist der erste Buchstabe auf der ersten Seite links oben zu finden, bei anderen Schriften ist das manchmal anders. Die Texte werden in Zeilen, Seiten, Absätze und so weiter gebrochen, aber das sind abstrakt gesprochen nicht unbedingt wesentliche Merkmale. Wie lang eine Zeile ist, hat für den Text an sich keine Bedeutung, genausowenig wie die Größe der Schrift oder die Größe der Seite wesentlich sind. Der Begriff der Zeichenkette abstrahiert deswegen völlig von den Fragen des Zeilenumbruchs: Ein Zeilenumbruch ist nur dem Medium Papier (oder Bildschirm) geschuldet und würde beim Vorlesen zum Beispiel völlig ignoriert werden. Was nach Abzug dieser Merkmale bleibt, ist tatsächlich eine Struktur, bei der es einen ersten Buchstaben gibt und einen letzten, und in der die Buchstaben linear aufeinanderfolgen. Diese lineare Abfolge kann man wie folgt kodieren.

Definition 6.1 (Alphabet, Wort, Zeichenkette) *Es sei A eine Menge, in diesem Zusammenhang auch das **Alphabet** genannt. Ein **Wort** oder **Zeichenkette** über A ist eine Funktion $f : \underline{n} \to A$ für eine natürliche Zahl n. Es ist n die **Länge** von f und wird auch mit $|f|$ bezeichnet.*

Diese Definition bedarf der Erklärung. Zunächst einmal ist ja $\underline{n} = \{\underline{0}, \underline{1}, \cdots, \underline{n-1}\}$, sodass f eine Abbildung von einer n-elementigen Menge in A ist. Wir sind es gewohnt, dass die Buchstaben des Wortes hinter einander geschrieben werden, etwa so: aabac. Zur Hervorhebung dieser Kett benutze ich Schrägstriche: /aabac/. Die Schrägstriche markieren Beginn und Ende der Zeichenkette. Wir betrachten dies hier lediglich als Abkürzung für die Funktion $f : \underline{5} \to A$ (wo A das gewöhnliche lateinische Alphabet ist) definiert durch

(6.1) $$f : \underline{0} \mapsto \mathtt{a}, \underline{1} \mapsto \mathtt{a}, \underline{2} \mapsto \mathtt{b}, \underline{3} \mapsto \mathtt{a}, \underline{4} \mapsto \mathtt{c}$$

Aufgrund unserer Definition einer Funktion bedeutet dies nunmehr, dass

(6.2) $$f = \{\langle \underline{0}, \mathtt{a}\rangle, \langle \underline{1}, \mathtt{a}\rangle, \langle \underline{2}, \mathtt{b}\rangle, \langle \underline{3}, \mathtt{a}\rangle, \langle \underline{4}, \mathtt{c}\rangle\}$$

Es wäre natürlich noch möglich, darauf hinzuweisen, dass dies ja laut Definition des Paares nichts anderes ist als

(6.3) $$f = \{\{\underline{0}, \{\underline{0}, \mathtt{a}\}\}, \{\underline{1}, \{\underline{1}, \mathtt{a}\}\}, \{\underline{2}, \{\underline{2}, \mathtt{b}\}\}, \{\underline{3}, \{\underline{3}, \mathtt{a}\}\}, \{\underline{4}, \{\underline{4}, \mathtt{c}\}\}\}$$

Und weiter kann man sagen, dass ja die Zahlen auch nichts anderes sind als Mengen — und so weiter. Das ist möglich, aber nicht hilfreich. Sämtliche Symbolisierungen, also etwa (6.1), (6.2) und (6.3), tun das gleiche: Sie legen f eindeutig fest. Welche wir wählen, hängt von uns ab. Und dann wählen wir natürlich diejenige, die uns am Besten vor Augen führt, wie f beschaffen ist.

Deswegen können wir letztendlich auch noch ganz andere Darstellungsformen wählen, da es ja nur darauf ankommt, dass wir uns zweifelsfrei mitteilen. Eine andere Visualisierung ist wie folgt. Wir stellen uns ein langes Band mit Zellen vor, welche jeweils einen einzigen Buchstaben tragen. Die Zellen sind abgezählt, die erste trägt die Nummer 0, die zweite die Nummer 1, und so weiter. Und dann sieht f wie folgt aus.

(6.4) $$\begin{array}{|c|c|c|c|c|} \hline \mathtt{a} & \mathtt{a} & \mathtt{b} & \mathtt{a} & \mathtt{c} \\ \hline \multicolumn{1}{c}{0} & \multicolumn{1}{c}{1} & \multicolumn{1}{c}{2} & \multicolumn{1}{c}{3} & \multicolumn{1}{c}{4} \end{array}$$

Ist $A \subseteq B$, so ist jedes Wort über A auch ein Wort über B. Der Wertebereich, das heißt das Alphabet, ist also teilweise unerheblich. f ist immer dann ein Wort über A, wenn A schon das ganze Bild von f enthält.

Es gibt ein ganz besonderes Wort, das **leere Wort**. Es ist die (eindeutig bestimmte) Funktion $\varepsilon : \underline{0} \to A$. ε ist das einzige Wort der Länge 0. (Es ist auch unabhängig vom Alphabet.) Wer die Definitionen aufmerksam liest, wird sehen, dass ε identisch mit der leeren Menge ist, also $\varepsilon = \varnothing$. Trotzdem verwenden wir für das leere Wort das Symbol ε.

Man beachte, dass es auf der Schreibmaschine auch noch die Leertaste gibt; wenn man diese drückt, erscheint kein Symbol, aber trotzdem rückt man im Text eine Stelle weiter. Im Computer hat allerdings das Leerzeichen einen Code, es wird behandelt wie ein Zeichen, das man sieht. Damit dies auch augenfällig wird, schreibt man gerne ␣, um die Gegenwart des Leerzeichens zu signalisieren. Und so ist zum Beispiel /`ab␣hier`/ eine Zeichenkette der Länge 7, welche aus zwei Worten besteht, /`ab`/ und /`hier`/, getrennt durch ein Leerzeichen. Es gibt also zwei verschiedene Leerheiten, wenn man so will. Die erste ist das leere Wort, die zweite das Wort /␣/, welche eine Funktion f von $\underline{1}$ nach A ist (wobei $␣ \in A$ vorausgesetzt wird) mit $f(\underline{0}) = ␣$).

Ich benutze Vektorpfeile ($\vec{x}$, $\vec{y}$ usw.), um Wörter (= Zeichenketten) zu bezeichnen. Sind nun $\vec{x} : \underline{m} \to A$ und $\vec{y} : \underline{n} \to A$ Zeichenketten über demselben Alphabet, so ist die **Verkettung** oder **Hintereinanderschreibung** $\vec{x} \cdot \vec{y}$ (oder einach nur $\vec{x}\vec{y}$) wie folgt definiert. Es ist $\vec{x} \cdot \vec{y} : \underline{m+n} \to A$, und zwar ist

$$(\vec{x} \cdot \vec{y})(\underline{k}) := \begin{cases} \vec{x}(\underline{k}) & \text{falls } k < m, \\ \vec{y}(\underline{k-m}) & \text{sonst.} \end{cases} \tag{6.5}$$

Dies ist eine Definition mit Hilfe von Fallunterscheidung. Sie sagt, dass $\vec{x} \cdot \vec{y}$ eine Funktion ist, die einer gegebenen Zahl, hier k genannt, einen Wert zuordnet. Dies tut sie auf zwei verschiedene Weisen: ist $k < m$, so tritt der erste Fall ein und der Wert wird als $\vec{x}(\underline{k})$ festgelegt. Ist $k \geq m$, so wird der Wert als $\vec{y}(\underline{k-m})$ festgelegt. Man beachte, dass nicht gleichzeitig der erste *und* der zweite Fall eintreten kann, die Definition also tatsächlich eindeutig ist. Man beachte ferner, dass wenn $k \geq m$, die zweite Definition sinnvoll ist: Denn dann ist $k - m \geq 0$, und, weil $k < m + n$, so ist $k - m < n$, und so ist $\vec{y}(\underline{k-m})$ tatsächlich definiert.

So ist zum Beispiel /`aabac`/ $\cdot$ /`bc`/ = /`aabacbc`/. Denn sei $g = \{\langle \underline{0}, \mathtt{b}\rangle, \langle \underline{1}, \mathtt{c}\rangle\}$ und f wie in (6.2). Dann ist $f \cdot g$ eine Funktion von $\underline{7}$ nach A, welche wie folgt definiert ist.

$$(6.6)\qquad \begin{aligned}(f\cdot g)(\underline{0}) &= f(\underline{0}) = \mathtt{a}\\ (f\cdot g)(\underline{1}) &= f(\underline{1}) = \mathtt{a}\\ (f\cdot g)(\underline{2}) &= f(\underline{2}) = \mathtt{b}\\ (f\cdot g)(\underline{3}) &= f(\underline{3}) = \mathtt{a}\\ (f\cdot g)(\underline{4}) &= f(\underline{4}) = \mathtt{c}\\ (f\cdot g)(\underline{5}) &= g(\underline{0}) = \mathtt{b}\\ (f\cdot g)(\underline{6}) &= g(\underline{1}) = \mathtt{c}\end{aligned}$$

Mit Hilfe von Zellen können wir das auch so notieren:

$$(6.7)\qquad \begin{array}{|c|c|c|c|c|}\hline \mathtt{a}&\mathtt{a}&\mathtt{b}&\mathtt{a}&\mathtt{c}\\\hline \multicolumn{1}{c}{0}&\multicolumn{1}{c}{1}&\multicolumn{1}{c}{2}&\multicolumn{1}{c}{3}&\multicolumn{1}{c}{4}\end{array} \cdot \begin{array}{|c|c|}\hline \mathtt{b}&\mathtt{c}\\\hline \multicolumn{1}{c}{0}&\multicolumn{1}{c}{1}\end{array} = \begin{array}{|c|c|c|c|c|c|c|}\hline \mathtt{a}&\mathtt{a}&\mathtt{b}&\mathtt{a}&\mathtt{c}&\mathtt{b}&\mathtt{c}\\\hline \multicolumn{1}{c}{0}&\multicolumn{1}{c}{1}&\multicolumn{1}{c}{2}&\multicolumn{1}{c}{3}&\multicolumn{1}{c}{4}&\multicolumn{1}{c}{5}&\multicolumn{1}{c}{6}\end{array}$$

Folgende Tatsache hebe ich eigens hervor. Es sei $|\vec{x}|$ die Länge der Zeichenkette $\vec{x}$. Ebenso sei für $a \in A$ $|\vec{x}|_a$ die Anzahl der Vorkommen von a in $\vec{x}$. (Das heißt, wir zählen, wie viele i existieren mit $\vec{x}(\underline{i}) = a$.)

Satz 6.2 $|\vec{x}\cdot\vec{y}| = |\vec{x}| + |\vec{y}|$. *Ferner ist* $|\vec{x}\cdot\vec{y}|_a = |\vec{x}|_a + |\vec{y}|_a$ *für jedes* $a \in A$.

Die erste Behauptung ist nach Definition der Verkettung richtig. Die zweite ist ebenfalls nicht schwer zu sehen.

Man beachte, dass $\vec{x}\cdot\vec{y}$ eigentlich immer definiert ist, auch wenn die Zielalphabete nicht übereinstimmen. Wie ich oben schon sagte, einigt man sich in diesem Fall darauf, dass stillschweigend das Alphabet erweitert wird. Ist also $\vec{x} : \underline{m} \to A$ und $\vec{y} : \underline{n} \to B$, so betrachten wir beide als Funktionen nach $A \cup B$. Dies ist im Übrigen kein Widerspruch zur Definition. Eine Funktion ist ja lediglich eine gewisse Menge von Paaren. Insofern ist $\vec{x}$ als Funktion von $\underline{m}$ nach A *dieselbe Menge* wie $\vec{x}$ als Funktion von $\underline{m}$ nach $A \cup B$.

Proposition 6.3 *Die Verkettung von Wörtern hat folgende Eigenschaften.*

① $\vec{x}\cdot\varepsilon = \varepsilon\cdot\vec{x} = \vec{x}$.

② $\vec{x}\cdot(\vec{y}\cdot\vec{z}) = (\vec{x}\cdot\vec{y})\cdot\vec{z}$.

③ *Ist* $\vec{x}\cdot\vec{z} = \vec{y}\cdot\vec{z}$, *so ist* $\vec{x} = \vec{y}$.

④ *Ist* $\vec{z}\cdot\vec{x} = \vec{z}\cdot\vec{y}$, *so ist* $\vec{x} = \vec{y}$.

Der Beweis ist einigermaßen langatmig und sei hier ausgelassen.

Definition 6.4 (Präfix, Suffix) *Es sei $\vec{x}$ ein Wort. $\vec{y}$ ist ein **Präfix** von $\vec{x}$, falls es ein $\vec{u}$ gibt mit $\vec{y} \cdot \vec{u} = \vec{x}$. $\vec{y}$ heißt **Postfix** von $\vec{x}$, falls es ein $\vec{u}$ gibt mit $\vec{x} = \vec{u} \cdot \vec{y}$. $\vec{y}$ heißt **Teilwort** von $\vec{x}$, falls es $\vec{u}$ und $\vec{v}$ gibt mit $\vec{x} = \vec{u} \cdot \vec{y} \cdot \vec{v}$.*

Sei wieder die Zeichenkette /aabac/ gegeben. Die folgenden Zeichenketten sind Präfixe: ε, /a/, /aa/, /aab/, /aaba/, sowie die Zeichenkette selbst, /aabac/. Die Postfixe sind /aabac/, /abac/, /bac/, /ac/, /c/ und ε. Teilworte sind außer den bisher genannten noch /ab/, /aba/, /b/ und /ba/.

In dem genannten Beispiel ist /a/ ein Teilwort das, wie wir sagen, drei Vorkommen hat, es tritt dreimal auf: am Anfang, an zweiter Stelle und an vierter Stelle. Wenn wir eine sehr lange Zeichenkette haben, dann kann es vorkommen, dass wir eigentlich gar nicht so sehr an einem Teilwort interessiert sind, sondern an der Frage, wo es denn auftritt. Wir wollen zum Beispiel in einem Text sämtliche Vorkommen eines Wortes durch ein anderes ersetzen, wie etwa in der Geschichte von Heinrich Böll, *Dr. Murkes gesammeltes Schweigen*, wo der Titelheld in der Tonaufzeichnung eines Vortrags jedes Vorkommen von „Gott" durch „jenes höhere Wesen, das wir verehren" ersetzen muss. Seine Aufgabe besteht darin (da eine komplette Neuaufnahme nicht infrage kommt), in der Tonspur diese Vorkommen aufzusuchen, um sie dann regelrecht herauszuschneiden und die Ersatzaufnahmen der Phrase „jenes höhere Wesen, das wir verehren" einzukleben. Da das Wort „Gott" mehrfach vorkommt, muss Dr. Murke denn auch mehrfach Ersatzaufnahmen parat haben, die er an die Stelle des Originals setzen kann, übrigens auch solche im Genitiv, Dativ und Akkusativ!

In unserem formalen Apparat müssen wir also noch Vorkommen definieren. Ist eine Zeichenkette $\vec{x} : \underline{m} \to A$ gegeben, so ist ein *Vorkommen* eines gegebenen Teilwortes $\vec{y}$ offenkundig eindeutig dadurch definiert, dass wir seinen Anfangspunkt benennen.

(6.8) $\mathtt{W}_0\mathtt{enn}\textvisiblespace\mathtt{h}_5\mathtt{inter}_{10}\textvisiblespace\mathtt{Fliegen}\textvisiblespace\mathtt{F}_{20}\mathtt{liegen}\textvisiblespace\mathtt{fli}_{30}\mathtt{egen,}\textvisiblespace\mathtt{flie}_{40}$
$\mathtt{gen}\textvisiblespace\mathtt{Fliege}_{50}\mathtt{n}\textvisiblespace\mathtt{Fliegen}\textvisiblespace_{60}\mathtt{nach.}$

Betrachten wir die Kette in (6.8) und das Wort /fliegen/. Dieses Wort hat zwei Vorkommen: eines beginnt bei Position 28, ein anderes bei Position 37. Das Wort /liegen/ hat 6 Vorkommen: bei 13, 21, 29, 38, 46 und 54. Das Wort /Fliegen/ hat 4 Vorkommen (12, 20, 45 und 53).

Wenn wir hingegen das Teilwort selber nicht benennen, so können wir es immer noch eindeutig bestimmen, indem wir nicht nur die Anfangsposition angeben sondern auch die Endposition. Ich gebe diese als Paar $\langle \underline{m}, \underline{n} \rangle$ von Zahlen an. Man beachte also, dass jetzt die Zeichenkette $\vec{x}$ immer noch gegeben ist, hingegen das Teilwortvorkommen nur noch ein Zahlenpaar ist, aus dem sich das Wort dann rekonstruieren lässt.

Definition 6.5 (Vorkommen von Zeichenketten) *Sei $\vec{x} : \underline{n} \to A$ eine Zeichenkette der Länge n. Dann ist ein Paar von natürlichen Zahlen $\langle \underline{i}, \underline{j} \rangle$ mit $i \leq j \leq n$ ein* ***Vorkommen*** *eines Teilwortes von $\vec{x}$. Das durch dieses Vorkommen definierte Wort ist $\vec{u} : \underline{j - i} \to A$ mit*

(6.9) $$\vec{u}(\underline{k}) := \vec{x}(\underline{k + i})$$

Zum Beispiel definiert $\langle \underline{1}, \underline{3} \rangle$ für f das Vorkommen eines Teilworts der Länge 2, welches bei $\underline{1}$ anfängt und bei $\underline{3}$ zu Ende ist. Es enthält also lediglich die Zeichen $f(\underline{1}) = \mathtt{a}$ und $f(\underline{2}) = \mathtt{b}$. Wir haben also ein Vorkommen des Teilworts /ab/.

(6.10) $$\begin{array}{|c|c|c|c|c|} \hline \mathtt{a} & \mathtt{a} & \mathtt{b} & \mathtt{a} & \mathtt{c} \\ \hline \end{array} \\ \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array}$$

Man beachte die Vorteile dieser Konvention. Ist $i = j$, so haben wir ein Vorkommen des leeren Wortes. Dieses hört im selben Moment auf, wie es begonnen hat. Hingegen sieht ein Wort aus einem Buchstaben anders aus, wie hier das einzige Vorkommen des Wortes /b/, $\langle \underline{2}, \underline{3} \rangle$:

(6.11) $$\begin{array}{|c|c|c|c|c|} \hline \mathtt{a} & \mathtt{a} & \mathtt{b} & \mathtt{a} & \mathtt{c} \\ \hline \end{array} \\ \begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \end{array}$$

In Figur 6.1 sind alle Vorkommon aufgelistet. Ein Präfix ist ein Teilwort, welches ein Vorkommen der Form $\langle \underline{0}, \underline{k} \rangle$ hat, ein Suffix ein Teilwort, welches ein Vorkommen der Form $\langle \underline{k}, \underline{m} \rangle$ hat, wo $\underline{m}$ die Länge der Zeichenkette ist. Man beachte, dass wir nicht von Präfixvorkommen reden, wohl aber von Teilwortvorkommen. Das hat einen wichtigen Grund. Ein gegebenes Wort kann gewiss mehrfach als Teilwort vorkommen, aber höchstens einmal als Präfix, und höchstens einmal als Postfix. Das liegt daran, dass beim Präfix die Anfangsposition feststeht: Es ist die 0. Geben wir noch das Teilwort $\vec{y}$ selber bekannt, so ist das Vorkommen eindeutig bestimmt: Es ist $\langle \underline{0}, \underline{n} \rangle$, wo n die Länge von $\vec{y}$ ist. Und genauso liegt das Vorkommen von $\vec{y}$ (wenn es existiert) eindeutig fest. Es ist $\langle \underline{m - n}, \underline{m} \rangle$.

Ich mache noch einige Ausführungen zu den Relationen, in denen Teilworte zueinander stehen. Zunächst eine Definition.

Abbildung 6.1: Die Teilwortvorkommen von /`aabac`/

	j	$\to$					
		0	1	2	3	4	5
i	0	ε	`a`	`aa`	`aab`	`aaba`	`aabac`
$\downarrow$	1		ε	`a`	`ab`	`aba`	`abac`
	2			ε	`b`	`ba`	`bac`
	3				ε	`a`	`ac`
	4					ε	`c`
	5						ε

Definition 6.6 (Träger) *Es sei $\vec{x}$ eine Zeichenkette der Länge n und $p = \langle \underline{i}, \underline{j} \rangle$ ein Teilwortwortvorkommen. Dann heißt die Menge $\underline{j} - \underline{i} := \{\underline{i}, \underline{i+1}, \cdots, \underline{j-1}\}$ der* ***Träger*** *von p, in Zeichen* $\mathrm{Tr}(p)$*.*

Die folgende Behauptung ist leicht zu sehen.

Lemma 6.7 *Es sei $\vec{x} : \underline{n} \to A$. Eine Teilmenge $S \subseteq \underline{n}$ ist genau dann Träger eines Teilwortes von $\underline{x}$, wenn es ein Intervall ist, das heißt, sind $\underline{i}, \underline{j} \in S$ und $i < k < j$, dann auch $\underline{k} \in S$.*

Die Anzahl der Elemente der Trägers gibt an, welche Länge das Wort hat, dessen Vorkommen es ist. Ist der Träger leer, so haben wir ein Vorkommen des leeren Wortes.

Es seien $p = \langle \underline{i}, \underline{j} \rangle$ und $q = \langle \underline{k}, \underline{\ell} \rangle$ Vorkommen von Teilworten.

1. p ist **vor** q falls $j \leq k$.
2. p und q sind **adjazent**, wenn $j = k$ oder $\ell = i$.
3. p und q **überlappen**, wenn p nicht vor q ist und q nicht vor p.

Betrachten wir dies etwas ausführlicher. Wir haben die Träger von p und q:

$$\mathrm{Tr}(p) = \{\underline{i}, \underline{i+1}, \cdots, \underline{j-1}\}, \mathrm{Tr}(q) = \{\underline{k}, \underline{k+1}, \cdots, \underline{\ell - 1}\} \tag{6.12}$$

Nehmen wir erst einmal den Fall, wo $\mathrm{Tr}(p)$ leer ist, das heißt, $i = j$. Dann ist p vor q, falls $i = j < k$; und p ist nach q, falls $\ell \leq i = j$. Also überlappt p mit q, falls $k < i = j < \ell$. Der Fall, wo $\mathrm{Tr}(q)$ leer ist, ist analog. Es gibt lediglich eine

Ausnahme: Sind $\mathrm{Tr}(p)$ und $\mathrm{Tr}(q)$ beide leer, so können wir sowohl sagen. dass p vor q als auch, dass p nach q ist. Schließlich sei der Fall betrachtet, wo $\mathrm{Tr}(p)$ und $\mathrm{Tr}(q)$ beide nicht leer sind. In diesem Fall überlappen p und q genau dann, wenn $\mathrm{Tr}(p) \cap \mathrm{Tr}(q) \neq \varnothing$. Denn sei $\mathrm{Tr}(p) \cap \mathrm{Tr}(q) = \varnothing$. Dann ist $\underline{j-1} \notin \mathrm{Tr}(q)$. Dies bedeutet $j \leq k$ (und damit, dass p vor q ist) oder $j > \ell$. Im zweiten Fall wissen wir zusätzlich, dass $\underline{i} \notin \mathrm{Tr}(q)$, also ist auch $i \geq \ell$, das heißt, q ist vor p. Die Umkehrung ist ebenso einfach.

Proposition 6.8 *Sind $p = \langle \underline{i}, \underline{j} \rangle$ und $q = \langle \underline{k}, \underline{\ell} \rangle$ Teilwortkommen von $\vec{x}$, die überlappen. Dann gilt:*

1. $\mathrm{Tr}(p) = \varnothing$ *und* $k < i = j < \ell$ *oder*

2. $\mathrm{Tr}(q) = \varnothing$ *und* $i < k = \ell < j$ *oder*

3. $\mathrm{Tr}(p) \cap \mathrm{Tr}(q) \neq \varnothing$.

In dem Fall, wo p und q adjazent sind, können wir die Verkettung von p und q definieren; dies ist wieder eine Teilwortvorkommen.

$$p \cdot q := \begin{cases} \langle \underline{i}, \underline{\ell} zu & \text{falls } p \text{ vor } q,\ p \text{ und } q \text{ adjazent.} \\ \text{undefiniert} & \text{sonst} \end{cases} \tag{6.13}$$

Definition 6.9 (Sprache) *Eine **Sprache über** A ist eine beliebige Menge von Wörtern über A. Die Menge aller Zeichenketten über A wird mit A^* bezeichnet. Eine beliebige Menge S von Zeichenketten heißt demnach Sprache, wenn es ein endliches A gibt derart, dass $S \subseteq A^*$.*

Diese Definition ist recht schlicht, und wir werden noch Gelegenheit haben, spezielle Sprachen zu studieren. Im Augenblick aber wollen wir es bei dieser Definition belassen und einen kurzen Blick auf die Frage werfen, wie viele Sprachen es denn nach dieser Definition so gibt.

Ich beende diesen Abschnitt noch mit einem Hinweis. Ich habe bisher stets die Notation $\underline{i}$ für natürliche Zahlen verwendet, sofern ich die Zahl als Menge benötigte. Im Verlaufe der Zeit werde ich diesen Unterschied aber nicht immer durchhalten, er ist auch für das Verständnis der Materie nicht erforderlich. (Und mehr Symbole zu haben ist auch nicht immer besser für das Verständnis.)

Übungen

[23] Geben Sie die Zeichenkette /`Kenner`/ als Funktion in Mengenschreibweise wieder. Welche Länge hat sie? Ist die Funktion (a) injektiv, (b) surjektiv?

[24] Geben Sie das Wort /`Wein`/ als Funktion an (wie in der ersten Aufgabe) und berechnen Sie explizit das Produkt mit der Zeichenkette /`kenner`/. Zeigen Sie dadurch vor allem die Indexverschiebung auf. (Frage: warum nehme ich nicht das Wort /`Kenner`/?)

[25] Geben Sie sämtliche (a) Präfixe, (b) Postfixe und (c) Teilworte des Wortes /`Kenner`/ an und benennen Sie die Vorkommen.

[26] Zerlegen Sie das Wort /`Kenner`/ auf alle möglichen Arten in eine Verkettung aus Präfix und Postfix. Wie viele gibt es davon?

Kapitel 7

Wie viele und welche Sprachen gibt es?

In diesem Abschnitt wollen wir einmal die Frage beleuchten, was denn eine formale Definition von Sprache leisten kann und was nicht. Sprachwissenschaft ist die Wissenschaft von der Sprache; welche Sprachen aber sind gemeint? Zuerst sind natürlich alle Sprachen gemeint, die von Menschen gesprochen werden. Dabei gibt es natürlich gewachsene (Deutsch, Finnisch, Baskisch und so weiter) und Kunstsprachen (Esperanto, Volapük). Daneben noch Sprachen, die nicht gesprochen werden, wie Gebärdensprachen, aber eben auch Computersprachen, sowie Fachsprachen. Neuerdings interessieren sich Sprachwissenschaftler auch für das Sprachvermögen von Tieren. Das erweitert die Definition um solche Sprachen, die von Tieren und eben nicht von Menschen gesprochen werden (oder was immer das im Fall von Vögeln heißen soll). Diese Aufzählung birgt schon einige Probleme. Noam Chomsky hat zum Beispiel wiederholt betont, natürliche Sprachen seien anders als formale Sprachen. Allerdings ist es nicht leicht, einen konkreten Unterschied zu benennen. Andere, zum Beispiel Richard Montague, haben keinen derartigen Unterschied gemacht. Es ist klar, dass es nur endlich viele Sprachen gibt und gegeben hat; es ist auch klar, dass es in Zukunft neue Sprachen geben wird, und diese werden anders sein als alle Sprachen vor ihnen. Wenn wir also fragen, welche Sprachen denn nun zu unserem Untersuchungsgebiet gehören, so können es wohl nicht einfach nur die existierenden sein. Die möglichen Sprachen sollten auch dazugehören. Das wirft natürlich die Frage auf: welche Sprachen sind denn nun möglich? Und wie viele mögliche Sprachen gibt es eigentlich?

Bevor wir die Frage beantworten können, müssen wir erst einmal definieren, was das denn eigentlich heißt, zu sagen, es gibt soundsoviele Dinge. Dazu erst einmal eine Definition.

Definition 7.1 (Mächtigkeit) *Es seien M und N Mengen. Dann hat M* ***mindestens die Mächtigkeit von*** *N, in Zeichen* $|M| \geq |N|$*, wenn es eine injektive Funktion von N nach M gibt. Wir sagen, M und N seien* ***gleich mächtig****, in Zeichen* $|M| = |N|$*, wenn es eine bijektive Funktion von M nach N gibt. Ist* $|M| \leq |N|$ *ohne dass* $|M| = |N|$*, so schreiben wir* $|M| < |N|$*.*

Der Hintergrund dieser Definition ist folgender. Wenn wir sagen, eine Menge M habe 4 Elemente und werden gebeten, dies zu zeigen, dann werden wir die Elemente einzeln vorzeigen und sagen: dies ist das erste Element, dies das zweite, dies das dritte, und dies das vierte. Und wenn dann keines mehr übrig ist, ist der Beweis erbracht. Ist zum Beispiel $M = \{\diamondsuit, \heartsuit, \spadesuit, \clubsuit\}$, so können wir sagen: $\clubsuit$ ist das erste, $\heartsuit$ das zweite, $\diamondsuit$ das dritte und $\spadesuit$ das vierte Element. Nun braucht man zum Abzählen nicht unbedingt Zahlen. Im Zweistromland hatte man Viehtreibern außer der Herde auch noch ein Kästchen aus Ton mitgegeben, in welchem genau so viele Tierfiguren enthalten waren, wie auch in der Herde bei der Abreise vorhanden waren. Wenn der Viehtreiber am Ende der Reise die Herde übergab, wurde das Kästchen zerbrochen und die Herde mittels der Figuren durchgezählt. Es wurde also zwischen der Menge der Figuren und den Tieren der Herde eine Bijektion hergestellt. Dabei war den Menschen auch damals intuitiv klar, dass es nicht darauf ankam, welche Figur welchem Tier zugeordnet wurde. Welcherart die Bijektion ist, ist für die Anzahl unerheblich; einzig wesentlich ist, dass es eine Bijektion überhaupt gibt. Blieben Tiere übrig, hatte der Viehtreiber mehr Tiere angeliefert, als ihm vorher übergeben waren. Blieben allerdings Figuren übrig, hatte der Viehtreiber ein Problem ...

In der Mengenlehre hat sich irgendwann herausgestellt, dass es für den Zweck des Zählens ganz bestimmte Mengen gibt, die man zum Ausschöpfen mittels einer Funktion nehmen kann, sogenannte Kardinalzahlen. Dazu weiter unten.

Natürlich ist es beim Abzählen unerheblich, welches Element wir zum ersten erklären. Die Ordnungszahlen werden den Elementen willkürlich gegeben, so wie auch unerheblich ist, welche Tierfigur zu welchem Tier passt (außer natürlich, wenn es sich um verschiedene Arten handelt, weil die ja nicht gleich teuer waren). Vom Standpunkt der obigen Definition werden wir sagen: wir haben also eine Zuordnung (= Funktion) zwischen Elementen der Menge M und den Zahlen. Und zwar nicht einfach irgendwelchen Zahlen. Sondern wir nehmen die ersten Zahlen, die wir finden können. Im normalen Leben beginnen wir mit 1, aber in der Mathematik (und Informatik) beginnt man mit 0. Und dann zählen wir wie gehabt weiter: 1, 2, 3, $\cdots$. Unsere Funktion F sieht zum Beispiel so aus:

$$F = \{\langle \underline{0}, \clubsuit\rangle, \langle \underline{1}, \heartsuit\rangle, \langle \underline{2}, \diamondsuit\rangle, \langle \underline{3}, \spadesuit\rangle\} \tag{7.1}$$

In der Tat ist sie auf den ersten 4 Zahlen definiert. Da nun auf der anderen Seite $\underline{4} = \{\underline{0}, \underline{1}, \underline{2}, \underline{3}\}$ ist, ist jetzt F eine bijektive Abbildung von $\underline{4}$ auf M. Und so kommen wir darauf, zu sagen, M habe 4 Elemente. Halten wir also fest:

> Eine Menge M hat die **Mächtigkeit** k, wenn es eine Bijektion von $\underline{k}$ auf M gibt.

Wieso ist nun diese Definition eindeutig? Kann es nicht sein, dass wir eine Bijektion $F : \underline{m} \to M$ und eine Bijektion $G : \underline{n} \to M$ finden, wo $m \neq n$ ist? Dazu lautet die Antwort: das ist unmöglich. Ich mache das wie folgt plausibel. Zunächst ist $G^{\smile}$ auch eine Funktion, weil G bijektiv ist; und dann ist $G^{\smile}$ auch eine Bijektion. Und schließlich ist $G^{\smile} \circ F : \underline{m} \to \underline{n}$ auch eine Bijektion. Es folgt also, dass $\underline{m}$ und $\underline{n}$ die gleiche Mächtigkeit haben. Das geht nicht, wie wir wissen. Und das kann man sogar beweisen!

Satz 7.2 (Cantor) *Für je zwei Mengen M und N gilt stets genau einer der drei Fälle: (a) $|M| < |N|$, (b) $|M| = |N|$, (c) $|M| > |N|$.*

Die Relation „ist gleichmächtig mit" ist eine Äquivalenzrelation. Zunächst ist jede Menge gleichmächtig mit sich selbst. Die Identitätsfunktion ist nämlich eine Bijektion der Menge auf sich. Ferner ist mit $f : M \to N$ auch $f^{\smile} : N \to M$ eine Bijektion, und mit $f : M \to N$ und $g : N \to P$ auch $g \circ f : M \to P$. Dies rechtfertigt die Schreibweise „$|M| = |N|$" für die Tatsache der Gleichmächtigkeit von M und N.

In der Mengenlehre definiert man anschließend sogenannte **Kardinalzahlen**. Diese haben die folgende Eigenschaft: sind α und β Kardinalzahlen, so ist $\alpha \in \beta$ genau dann wenn $|\alpha| < |\beta|$ und $\alpha = \beta$ genau dann, wenn $|\alpha| = |\beta|$. Ich gebe hier kommentarlos eine Definition wieder. Die Konsequenzen dieser Definition zu verstehen, ist nicht leicht, weswegen sie im Folgenden auch keine Rolle spielen wird.

Definition 7.3 *Eine Menge α heißt **Ordinalzahl**, wenn sie durch $\in$ linear und transitiv geordnet ist. Eine **Kardinalzahl** ist eine Ordinalzahl α, für die zusätzlich gilt: für alle $\beta \in \alpha$ ist $|\beta| < |\alpha|$.*

Die endlichen Kardinalzahlen sind die Mengen $\underline{n}$, die wir schon definiert haben. Die erste unendliche Kardinalzahl heißt $\aleph_0$. Diese ist definiert durch

(7.2) $\aleph_0 := \{\underline{0}, \underline{1}, \underline{2}, \underline{3}, \cdots\}$

Gleichzeitig ist dies auch die erste unendliche Ordinalzahl und wir deswegen auch mit ω bezeichnet. Als Menge sind diese allerdings identisch. Eine Menge, welche gleichmächtig ist mit $\aleph_0$ heißt **abzählbar unendlich**. (In vielen Büchern sagt man nur **abzählbar**, aber ich halte das für verwirrend.) Ich gebe hier ein Beispiel einer Ordinalzahl an, welche keine Kardinalzahl ist:

(7.3) $\aleph_0 + 1 := \aleph_0 \cup \{\aleph_0\}$

Dies ist die Menge $\aleph_0$ vermehrt um ein einziges Element.

Satz 7.4 $|\aleph_0 + 1| = |\aleph_0|$.

Beweis. Es sei $f : \aleph_0 \to \aleph_0 + 1$ die folgende Funktion: $f(0) := \aleph_0$, und $f(\underline{n+1}) := \underline{n}$. Dies Funktion ist bijektiv. Zunächst einmal ist sie injektiv. Denn ist $f(x) = f(y)$, so ist $f(x)$ entweder endlich, d.h. $f(x) = \underline{n}$ für ein n, und so $x = \underline{n+1}$ und deswegen auch $y = \underline{n+1} = x$. Oder $f(x) = \aleph_0$, also $x = \underline{0} = y$. Es ist f auch surjektiv, wie man leicht sieht. ⊣

Ich gebe noch einen überraschenden Satz an. Die Kardinalzahl von $M \times N$ ist im endlichen Fall das Produkt der Kardinalzahlen von M und N. Dies ist größer als die Kardinalitäten von M und N, wenn diese mindestens zwei Elemente haben. Im Unendlichen ist dies aber nicht so.

Satz 7.5 $|\aleph_0 \times \aleph_0| = |\aleph_0|$.

Beweis. Wir brauchen eine Bijektion von der Menge der Zahlenpaare $\langle \underline{m}, \underline{n} \rangle$ auf die Menge der natürlichen Zahlen. Wir gehen dazu die Zahlenpaare wie folgt ab. Wir zählen erst die Paare ab, deren Summe 0 ist, dann diejenigen mit Summe 1, dann diejenigen mit Summe 2 und so weiter. Die Paare mit Summe $\underline{n}$ zählen wir so auf: $\langle \underline{n}, \underline{0} \rangle$, $\langle \underline{n-1}, \underline{1} \rangle$, $\langle \underline{n-2}, \underline{2} \rangle$, und so weiter bis $\langle \underline{0}, \underline{n} \rangle$.

(7.4) $\langle \underline{0}, \underline{0} \rangle, \langle \underline{1}, \underline{0} \rangle, \langle \underline{0}, \underline{1} \rangle, \langle \underline{2}, \underline{0} \rangle, \langle \underline{1}, \underline{1} \rangle, \langle \underline{0}, \underline{2} \rangle, \langle \underline{3}, \underline{0} \rangle, \langle \underline{2}, \underline{1} \rangle, \langle \underline{1}, \underline{2} \rangle, \langle \underline{0}, \underline{3} \rangle, \cdots$

Dies ist offenkundig eine Zählreihe, und wir setzen $f(\langle \underline{n}, \underline{m} \rangle)$ gerade die Ordnungsnummmer in der Zählreihe. ⊣

Satz 7.6 *Die Menge der endlichen Teilmengen von* $\aleph_0$ *ist abzählbar.*

Beweis. Jede endliche Teilmenge E von $\aleph_0$ hat ein größtes Element $e^\star$. Ist E nicht nicht leer, so ist $E = \{e^\star\} \cup E'$, wo E' eine beliebige Teilmenge von $e^\star = \{\underline{0}, \underline{1}, \cdots, e^\star - 1\}$ ist. Davon gibt es $2^{e^\star}$ viele. Wir beginnen also mit $\varnothing$ und zählen dann die Mengen mit $e^\star = \underline{0}$ auf, dann diejenigen mit $e^\star = \underline{1}$, und so weiter.

$$(7.5) \qquad \varnothing, \{\underline{0}\}, \{\underline{1}\}, \{\underline{0}, \underline{1}\}, \{\underline{2}\}, \{\underline{2}, \underline{0}\}, \{\underline{2}, \underline{1}\}, \{\underline{2}, \underline{0}, \underline{1}\}, \cdots$$

Damit wird jede Teilmenge genau einmal gezählt. ⊣

Um eine Menge zu bekommen, die echt größer ist als $\aleph_0$, muss man also schon etwas härter arbeiten. Das **Kontinuum** ist die Menge aller Teilmengen von $\aleph_0$ (also $\wp(\aleph_0)$). Dies ist zwar keine Kardinalzahl, aber man kann zeigen, dass diese Menge echt mächtiger ist als $\aleph_0$. Ich führe den Beweis hier vor. Jede Teilmenge S von $\aleph_0$ stellen wir als eine unendlich lange Folge f von Nullen und Einsen dar. Diese Folge ist so beschaffen, dass an der Stelle n dieser Folge eine 1 steht, wenn $\underline{n} \in S$, andernfalls steht eine Null. Die Menge der geraden Zahlen wird also zu der Folge

$$(7.6) \qquad 10101010101010\cdots$$

(Bedenken Sie, die erste Stelle heißt 0, und 0 ist gerade.) Die Menge der Primzahlen wird übersetzt in

$$(7.7) \qquad 00110101000101\cdots$$

Offenkundig ist die Korrespondenz zwischen Teilmengen von $\aleph_0$ und Folgen exakt (= bijektiv). Wir nehmen nun an, es gebe ein Aufzählung der Teilmengen von $\aleph_0$ in der Form S_0, S_1, S_2 und so weiter (also eine Bijektion von $\aleph_0$ auf $\wp(\aleph_0)$). Diese stellen wir nun als eine Aufzählung sämtlicher 0-1-Folgen dar, das heißt, wir haben eine Aufzählung f_0, f_1, f_2 sämtlicher 0-1-Folgen. Ich definiere nun eine Folge f^* wie folgt:

$$(7.8) \qquad f^*(i) := \begin{cases} 1 & \text{falls } f_i(i) = 0 \\ 0 & \text{falls } f_i(i) = 1 \end{cases}$$

Ich behaupte, dass die Folge f^* in dieser Aufzählung nicht vorkommt. Denn wenn sie dort vorkommt, so hat sie eine Nummer, sagen wir i; dann ist also $f^* = f_i$. Nehmen wir an, $f_i(i) = 0$; dann ist aber nach Definition $f^*(i) = 1$. Ist andererseits $f_i(i) = 1$, so ist $f^*(i) = 0$. Das ist aber ein Widerspruch.

Dieser Beweis (oder zumindest seine Idee) stammt von Georg Cantor und ist ein Glanzstück der Mathematik. Entgegen der intuitiven Idee, dass Unendlich einfach nur Unendlich ist und jenseits des Endlichen keine Größenunterschiede mehr existieren, hat Cantor zweifelsfrei zeigen können, dass es unendliche Mengen gibt, die echt größer sind als andere unendliche Mengen. Es gilt sogar noch mehr. Keine Menge kann ihre eigene Potenzmenge ausschöpfen.

Satz 7.7 (Cantor) *Für jede Menge M gilt $|M| < |\wp(M)|$.*

Der Beweis ist wie folgt. Angenommen, es existiert eine bijektive Abbildung $g : M \to \wp(M)$. Dann definieren wir folgende Menge: $H := \{k : k \notin g(k)\}$. Dies ist eine Teilmenge von M. Also existiert ein k^* mit $g(k^*) = H$. Ist nun $k^* \in H$, so muss, nach Definition von H, $k^* \notin g(k^*) = H$ sein. Ist aber $k^* \notin H$, so ist wiederum nach Definition $k^* \in g(k^*) = H$. Widerspruch. Also existiert kein k^* mit $g(k^*) = H$, und so ist g nicht surjektiv, entgegen der Annahme.

Ich merke noch an, dass Cantor vermutet hat, dass es keine Menge K gibt mit $\aleph_0 < |K| < |\wp(\aleph_0)|$. Dies war lange offen und hieß die Kontinuumshypothese. Kurt Gödel konnte zeigen, dass diese Hypothese immerhin widerspruchsfrei ist, das heißt, dass man nicht zeigen kann, dass sie falsch ist. Paul Cohen hat einige Zeit später gezeigt, dass man ebenfalls nicht zeigen kann, dass sie wahr ist. Diese Hypothese ist also, wie man sagt, unabhängig: weder kann man zeigen, dass sie falsch ist, noch kann man zeigen, dass sie falsch ist. Wie gesagt, die Tatsache, dass die Kontinuumshypothese weder beweisbar noch widerlegbar ist ihrerseits beweisbar. Mathematik kommt manchmal zu bizarren Ergebnissen.

Ich will hier noch ein wenig über Kardinalzahlen sagen. Diese sind gewissermaßen die Messzahlen. Die Größe einer Menge wird durch eine Kardinalzahl beschrieben. Mit diesen kann man sogar rechnen. Die einfachste Definition ist die des Produkts. Das Produkt zweier Kardinalzahlen ist diejenige Kardinalzahl, welche gleichmächtig ist zu dem kartesischen Produkt der Kardinalzahlen. Ich gebe dazu ein Beispiel. Die Menge $\underline{2} \times \underline{3}$ hat, wie man leicht sieht, 6 Elemente.

(7.9) $\{\langle\underline{0},\underline{0}\rangle, \langle\underline{0},\underline{1}\rangle, \langle\underline{0},\underline{2}\rangle, \langle\underline{1},\underline{0}\rangle, \langle\underline{1},\underline{1}\rangle, \langle\underline{1},\underline{2}\rangle\}$

Und das bedeutet nichts anderes, als dass eine Bijektion mit der Menge $\underline{6}$ existiert. Hier ist eine von ihnen (es gibt mehrere).

(7.10) $\{\langle\underline{0},\langle\underline{0},\underline{0}\rangle\rangle, \langle\underline{1},\langle\underline{0},\underline{1}\rangle\rangle, \langle\underline{2},\langle\underline{0},\underline{2}\rangle\rangle, \langle\underline{3},\langle\underline{1},\underline{0}\rangle\rangle, \langle\underline{4},\langle\underline{1},\underline{1}\rangle\rangle, \langle\underline{5},\langle\underline{1},\underline{2}\rangle\rangle\}$

Bei der Addition muss man etwas vorsichtiger sein. Die Addition von Kardinalzahlen wird wie folgt definiert. Hat M die Mächtigkeit κ und N die Mächtigkeit λ, *und ist M disjunkt zu N*, so hat $M \cup N$ die Mächtigkeit $\kappa + \lambda$. Im allgemeinen Fall gilt nämlich

$$|M \cup N| = |M| + |N| - |M \cap N| \tag{7.11}$$

M ist genau dann disjunkt zu N, wenn $M \cap N = \varnothing$, d.h. wenn $|M \cap N| = 0$.

Zunächst einmal beantworte ich die Frage danach, wie viele Wörter es gibt. Wir gehen davon aus, dass A stets endlich ist.

Proposition 7.8 *Es sei A endlich und nicht leer. Dann gibt es abzählbar unendlich viele Wörter über A.*

Beweis. Wir zählen die Wörter wie folgt auf. Zuerst kommen die Wörter der Länge 0, dann die der Länge 1, dann der Länge 2, und so weiter. Jedes Wort kommt offensichtlich in dieser Liste genau einmal vor. Die Funktion ist dann definiert als diejenige, die zu jeder Ordnungsnummer i das $i$te Element der Liste ausgibt. Dies ist eine Bijektion zwischen den natürlichen Zahlen und den Wörtern. ⊣

Ist $A = \{\mathtt{a}, \mathtt{b}\}$, so zählen wir wie folgt:

(7.12)

0	1	2	3	4	5	6	7	8	9	10
ε	a	b	aa	ab	ba	bb	aaa	aab	aba	abb

Damit ist geklärt:

Proposition 7.9 *Die Menge der Sprachen über einem endlichen, nichtleeren Alphabet hat die Mächtigkeit des Kontinuums.*

Dies finden einige gut, andere schlecht. Terence Langendoen und Paul Postal haben ein Buch geschrieben, in welchem nicht alleine behauptet wird, dass es (mindestens!) kontinuierlich viele Sprachen gibt, sondern auch, dass natürliche Sprachen unendlich lange Wörter enthalten. Das zweite folgt aus dem ersten, wenn wir annehmen, dass Alphabete endlich sind. (Langendoen und Postal gehen sogar so weit zu behaupten, dass natürliche Sprachen selbst keine Mengen sind.) Die radikale Gegenposition ist, dass Sprachen endlich sind, und es somit nur abzählbar viele Sprachen geben kann. Die allgemeine Position ist eher die, dass Sprachen zwar nicht endlich sein müssen, wohl aber eine endliche Beschreibung von ihnen existieren muss. In diesem Fall gibt es wohl nur abzählbar viele Sprachen. Intuitiv liegt das daran, dass man die endlichen Beschreibungen alle abzählen kann. Im Folgenden werden wir Sprachklassen definieren, welche nur abzählbar viele Sprachen enthalten. Nach dem oben Gesagten muss es dann immer Sprachen geben, die aus dieser Klasse herausfallen.

Eine Art der Beschreibung von Sprachen werde ich im nächsten Kapitel geben: Sprachen haben eine Grammatik. Auch wenn der Begriff der Grammatik sehr weit (sogar zu weit) gefasst ist, schränkt er doch schon den Begriff ein. Denn Grammatiken sind endlich. Und die endlichen Beschreibungen kann man abzählen. Angesichts des oben Gesagten habe wir folgendes Ergebnis. Die Bedingung, dass natürliche Sprachen eine Grammatik haben müssen, die sie erzeugt, ist *eine echte Einschränkung*. Denn da die Grammatiken abzählbar sind die Sprachen aber nicht, gibt es offenkundig Sprachen *ohne Grammatik*.

Übungen

[27] Zeigen Sie, dass aus $|M| \leq |N|$ und $|N| \leq |P|$ folgt, dass $|M| \leq |P|$. Die Relation $\leq$ ist somit transitiv. *Hinweis.* Offenkundig muss man dazu zeigen, dass die Verkettung von injektiven Funktionen wieder injektiv ist.

[28] Es sei folgende lineare Ordnung auf A^* gegeben. Wir fixieren eine lineare Ordnung $<$ auf A. $\vec{x}\, L\, \vec{y}$ sei genau dann der Fall, wenn ein $\vec{x}$ ein echtes Präfix von $\vec{y}$ ist oder ein i existiert, sodass (a) $\vec{x}(i) < \vec{y}(i)$, und (b) für alle $j < i$ ist $\vec{x}(j) = \vec{y}(j)$. Dies ist die sogenannte *lexikographische Ordnung*. Diese hängt selbstverständlich von $<$ ab. Zeigen Sie, dass diese Ordnung transitiv ist.

[29] Zeigen Sie, dass die lexikographische Ordnung linear ist.

[*30] Zeigen, dass es eine unendlich absteigende Folge von Zeichenketten mittels L gibt. Das bedeutet, dass $\vec{x}_i$, $i \in \mathbb{N}$, derart, dass für alle i gilt $\vec{x}_{i+1}\, L\, \vec{x}_i$.

[31] Es gibt Ausdrücke in der Sprache, die aus inhärenten Gründen keinen Gegenstand bezeichnen können. Dass dies so ist, hängt mit einer eigentümlichen Beziehung zwischen Sprache und Bedeutung zusammen. Betrachten wir die folgende Eigenschaft von natürlichen Zahlen: „kann nicht durch weniger als siebzehn Silben beschrieben werden".

[*32] Zeigen Sie, dass es keine Sprache gibt, bei der jede Menge natürlicher Zahlen eine Beschreibung besitzt. Dies bedeutet, dass natürliche Sprachen als Benennungssysteme für die Dinge unserer Gedankenwelt notwendig unvollständig sind, weil es immer Dinge gibt, die wir nicht benennen können. Das Gleiche gilt auch, wenn wir annehmen, dass alle reellen Zahlen existieren.

Kapitel 8

(Formale) Grammatiken

Ein zentraler linguistische Begriff ist der der *Grammatik*. Dabei hat das, was in der Theorie der formalen Sprachen als Grammatik bezeichnet wird, mit einer Grammatik im normalen Sprachgebrauch sehr wenig gemeinsam. Deswegen werde ich auch gelegentlich von *formaler Grammatik* reden. Eine formale Grammatik ist ein Erzeugungsprozess, mit dessen Hilfe man aus zum Teil künstlich aussehenden Zeichenreihen neue Zeichenreihen erzeugen kann. Am Ende dieses Prozesses steht immer eine Zeichenreihe aus der eigentlich zu erzeugenden Sprache. Das bedeutet: Die Grammatik erzeugt die zu betrachtende Sprache, indem sie einen Prozess beschreibt, in welchem Zeichenketten immer wieder verändert werden. Dies soll hier genauer erläutert werden.

Definition 8.1 (Ersetzungsregel) *Eine **Ersetzungsregel** ist ein Paar $\rho = \langle \vec{x}, \vec{y} \rangle$ von Zeichenketten. Wir schreiben ρ in der Form $\vec{x} \to \vec{y}$. Wir sagen, ρ **erlaube den Übergang von** $\vec{u}$ **nach** $\vec{v}$, falls es Wörter $\vec{w}_1$ und $\vec{w}_2$ gibt, derart, dass $\vec{u} = \vec{w}_1 \cdot \vec{x} \cdot \vec{w}_2$ und $\vec{v} = \vec{w}_1 \cdot \vec{y} \cdot \vec{w}_2$. Wir schreiben $\vec{u} \to_\rho \vec{v}$.*

Ich weise auf die Feinheit in der Schreibweise hin. $\vec{x} \to \vec{y}$ ist eine *Regel*, die die Ersetzung von $\vec{x}$ durch $\vec{y}$ beinhaltet; $\vec{x} \to_\rho \vec{y}$ (mit angehängtem ρ) ist eine *Aussage*; sie besagt, dass die Regel ρ erlaubt, aus $\vec{x}$ die Zeichenkette $\vec{y}$ zu machen. Dies bedeutet, dass die Regel ρ die Ersetzung eines beliebigen Vorkommens von $\vec{x}$ in einem Wort durch $\vec{y}$ erlaubt.

Ich gebe ein Beispiel. Es sei $\rho = \langle \mathtt{aa}, \mathtt{ba} \rangle$. Dies notieren wir auch $\mathtt{aa} \to \mathtt{ba}$. Sei nun das Wort /caaba/ gegeben. Dieses Wort besitzt ein einziges Vorkommen von /aa/, /c<u>aa</u>ba/.

(8.1) | c | a | a | b | a |

Wir haben damit $\vec{w}_1 = \mathtt{c}$ (hier olivgrün gezeichnet) und $\vec{w}_2 = \mathtt{ba}$ (in Blau). Es ist $\vec{x} = \vec{w}_1 \cdot \mathtt{aa} \cdot \vec{w}_2$. Wenden wir die Regel an, so wird das Vorkommen von /aa/ (in Rot) herausgetrennt und durch /ba/ ersetzt.

(8.2)
$$\begin{array}{c}\begin{array}{|c|c|c|c|c|}\hline \mathtt{c} & \mathtt{a} & \mathtt{a} & \mathtt{b} & \mathtt{a}\\\hline\end{array}\\ \downarrow \\ \begin{array}{|c|c|c|c|c|}\hline \mathtt{c} & \mathtt{b} & \mathtt{a} & \mathtt{b} & \mathtt{a}\\\hline\end{array}\end{array}$$

Nehmen wir eine andere Zeichenkette, /adaaab/. Hier haben wir zwei Vorkommen von /aa/:

(8.3)
$$\begin{array}{|c|c|c|c|c|c|}\hline \mathtt{a} & \mathtt{d} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \mathtt{b}\\\hline\end{array} \qquad \begin{array}{|c|c|c|c|c|c|}\hline \mathtt{a} & \mathtt{d} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \mathtt{b}\\\hline\end{array}$$

Je nachdem, welches Vorkommen wir wählen, bekommen wir ein verschiedenes Ergebnis:

(8.4)
$$\begin{array}{c}\begin{array}{|c|c|c|c|c|c|}\hline \mathtt{a} & \mathtt{d} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \mathtt{b}\\\hline\end{array}\\ \downarrow \\ \begin{array}{|c|c|c|c|c|c|}\hline \mathtt{a} & \mathtt{d} & \mathtt{b} & \mathtt{a} & \mathtt{a} & \mathtt{b}\\\hline\end{array}\end{array} \qquad \begin{array}{c}\begin{array}{|c|c|c|c|c|c|}\hline \mathtt{a} & \mathtt{d} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \mathtt{b}\\\hline\end{array}\\ \downarrow \\ \begin{array}{|c|c|c|c|c|c|}\hline \mathtt{a} & \mathtt{d} & \mathtt{a} & \mathtt{b} & \mathtt{a} & \mathtt{b}\\\hline\end{array}\end{array}$$

Aus diesem Grunde ist es nicht richtig zu sagen, man wende die Regel auf eine Zeichenkette an; sondern man muss sagen, dass man sie auf ein *Vorkommen* des Wortes in der Zeichenkette anwendet. Je nachdem, welches Vorkommen man benennt, bekommt man ein verschiedenes Ergebnis.

Dass die Länge der Zeichenkette gleich bleibt, ist nur deswegen so, weil die ersetzende Zeichenkette die gleiche Länge hat wie die, die ersetzt wird. Für die Regel $\langle \mathtt{aa}, \mathtt{c}\rangle$ gilt dies nicht.

(8.5)
$$\begin{array}{c}\begin{array}{|c|c|c|c|c|c|}\hline \mathtt{a} & \mathtt{d} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \mathtt{b}\\\hline\end{array}\\ \downarrow \\ \begin{array}{|c|c|c|c|c|}\hline \mathtt{a} & \mathtt{d} & \mathtt{c} & \mathtt{a} & \mathtt{b}\\\hline\end{array}\end{array} \qquad \begin{array}{c}\begin{array}{|c|c|c|c|c|c|}\hline \mathtt{a} & \mathtt{d} & \mathtt{a} & \mathtt{a} & \mathtt{a} & \mathtt{b}\\\hline\end{array}\\ \downarrow \\ \begin{array}{|c|c|c|c|c|}\hline \mathtt{a} & \mathtt{d} & \mathtt{a} & \mathtt{c} & \mathtt{b}\\\hline\end{array}\end{array}$$

Dies ist eine einfache Konsequenz aus dem folgenden Satz.

Satz 8.2 *Es sei $\rho = \vec{x} \to \vec{y}$. Ferner seien $\vec{u}$ und $\vec{v}$ Zeichenketten mit $\vec{u} \to_\rho \vec{v}$. Dann ist $|\vec{v}| = |\vec{u}| + |\vec{y}| - |\vec{x}|$.*

Zum Beweis merke ich an, dass $\vec{u} \to_\rho \vec{v}$ bedeutet, dass $\vec{w}_1$ und $\vec{w}_2$ existieren mit $\vec{u} = \vec{w}_1 \cdot \vec{x} \cdot \vec{w}_2$ und $\vec{v} = \vec{w}_1 \cdot \vec{y} \cdot \vec{w}_2$. Daraus folgt

(8.6)
$$|\vec{u}| = |\vec{w}_1| + |\vec{x}| + |\vec{w}_2|, \qquad |\vec{v}| = |\vec{w}_1| + |\vec{y}| + |\vec{w}_2|$$

Daraus folgt die Behauptung, denn jetzt ist

(8.7) $$|\vec{v}| = |\vec{w}_1| + |\vec{y}| + |\vec{w}_2| = (|\vec{w}_1| + |\vec{x}| + |\vec{w}_2|) - |\vec{x}| + |\vec{y}| = |\vec{u}| + |\vec{y}| - |\vec{x}|$$

Das Längeninkrement $|\vec{y}|-|\vec{x}|$ bei der Anwendung der Regel hängt also nur von der Regel ab. Von besonderer Bedeutung sind Regeln, bei denen $|\vec{y}| \geq |\vec{x}|$. Bei ihnen kann die Zeichenkette nach der Anwendung nicht kürzer werden.

Ich weise noch auf zwei Spezialfälle hin. Der erste ist eine Regel der Form $\langle \vec{x}, \varepsilon \rangle$. Diese bringt ein Vorkommen von $\vec{x}$ zum Verschwinden. So etwas nennt man eine **Tilgung**. Sei zum Beispiel die Regel $\mathtt{a} \to \varepsilon$ auf /aba/ angewendet, so bekommen wir wahlweise /ba/ oder /ab/, je nachdem, welches Vorkommen von /a/ wir ausgewählt haben. Der umgekehrte Fall ist die **Einfügung**: Das ist eine Regel der Form $\langle \varepsilon, \vec{x} \rangle$. Ein Beispiel soll illustrieren, wie diese Regel sich auswirkt. Das einfachste Beispiel ist die Regel $\langle \varepsilon, \mathtt{x} \rangle$. Diese fügt irgendwo den Buchstaben /x/ ein. Damit kann man aus der Zeichenkette /aaa/ wahlweise /xaaa/, /axaa/, /aaxa/ oder /aaax/ erzeugen. Ich erläutere das am ersten Beispiel. Das leere Wort hat in der Zeichenkette /aaa/ das Vorkommen $\langle 0, 0 \rangle$, also ganz am Anfang. Ersetzen wir es durch /x/, so steht nun /x/ vor allen Buchstaben: Wir bekommen /xaaa/.

Sei zum Beispiel $\rho = \langle \mathtt{a}, \mathtt{ä} \rangle$. Dann erlaubt ρ den Übergang von /Vater/ zu /Väter/. (Setze dazu $\vec{w}_1 := \mathtt{V}$ und $\vec{w}_2 = \mathtt{ter}$.) Es erlaubt aber nicht den Übergang von /Mutter/ zu /Mütter/, und auch nicht den Übergang von /Saal/ zu /Sääl/, weil die Ersetzung nur einmal vorgenommen werden darf. Allerdings können wir die Anwendung einer Regel wiederholen. Wir definieren nun wie folgt.

(8.8) $$\vec{x} \to_\rho^0 \vec{y} :\Leftrightarrow \vec{x} = \vec{y}$$

(8.9) $$\vec{x} \to_\rho^{n+1} \vec{y} :\Leftrightarrow \text{es existiert ein } \vec{z} \text{ mit } \vec{x} \to_\rho^n \vec{z} \to_\rho \vec{y}$$

(8.10) $$\vec{x} \to_\rho^* \vec{y} :\Leftrightarrow \text{es existiert ein } n \in \mathbb{N} \text{ mit } \vec{x} \to_\rho^n \vec{y}$$

Es ist dann zum Beispiel

(8.11) $$\mathtt{Saal} \to_\rho^2 \mathtt{Sääl}$$

Denn wir haben $\mathtt{Saal} \to_\rho \mathtt{Saäl} \to_\rho \mathtt{Sääl}$ (oder auch $\mathtt{Saal} \to_\rho \mathtt{Säal} \to_\rho \mathtt{Sääl}$). Falls wir nun eine Regel mehrfach anwenden können, so kann das Ergebnis sehr verschieden vom Ausgangspunkt sein. Wir sagen, eine ρ-**Ableitung** ist eine Folge von Zeichenketten $\vec{x}_0, \vec{x}_1, \vec{x}_2, \cdots, \vec{x}_n$ derart, dass jeweils $\vec{x}_i \to_\rho \vec{x}_{i+1}$; n ist die **Länge** der Ableitung. Es gilt also $\vec{x} \to_\rho^n \vec{y}$ immer dann, wenn es eine Ableitung der Länge

n von $\vec{y}$ aus $\vec{x}$ gibt. Hier ist zum Beispiel eine ρ-Ableitung für die obenstehende Regel ρ.

(8.12) $\langle$`Salamander`, `Sälamander`, `Sälamänder`, `Sälämänder`$\rangle$

(Jedes Anfangsstück einer Ableitung ist auch eine Ableitung; so ist etwa die Folge $\langle$`Salamander`, `Sälamander`, `Sälamänder`$\rangle$ auch eine Ableitung. Man *muss* also eine Regel nicht unbedingt anwenden, aber man *darf* es.)

Ich bringe noch ein instruktives und realistischeres Beispiel. Der Plural im Englischen wird in den meisten Fällen durch Anhängen des Buchstaben /`s`/ gebildet. Dies legt nahe, die Regel $\langle\varepsilon,$ `s`$\rangle$ zugrunde zu legen. Allerdings kann diese auf jede Zeichenkette angewendet werden und erlaubt, aus /`dog`/ nicht nur /`dogs`/ sondern auch /`sdog`/, /`dsog`/ und /`dosg`/ zu bilden, je nachdem, welches Vorkommen der leeren Zeichenkette wir „ersetzen". Natürlich wollen /`s`/ nur am Ende anhängen. Deswegen formuliere ich jetzt die Regel ein wenig anders:

(8.13) $\sqcup \rightarrow \mathtt{s}\sqcup$

Die Idee ist, dass ein Wort mit einem Leerzeichen aufhört. Daran können wir im Schriftbild erkennen, wo das /`s`/ angehängt werden muss. Dies ist zugegeben immer noch eine Vereinfachung, weil am Wortende noch andere Dinge stehen dürfen wie etwa Satzzeichen. Ich will diese Tatsache aber erst einmal ignorieren. (Komplizierter geht es immer!)

Für diese Regel gibt es noch eine alternative Schreibweise, nämlich

(8.14) $\varepsilon \rightarrow \mathtt{s}/\_\_\sqcup$

Diese Regel besagt, dass die Ersetzung von ε durch /`s`/ eingeschränkt wird durch eine Bedingung an den *Kontext*. Der Unterstrich bezeichnet hier das zu ersetzende Vorkommen (in diesem Fall des leeren Worts) und das nachfolgende /$\sqcup$/ besagt, dass auf dieses Vorkommen des leeren Worts ein Leerzeichen folgen muss. Es gibt aber nur ein Vorkommen des leeren Worts, auf das das Leerzeichen folgt, und dies ist genau am Ende des Wortes. Dies entspricht also genau der Ersetzung $\sqcup \rightarrow \mathtt{s}\sqcup$.

Bevor wir weitermachen, erst einmal eine formale Definition des Begriffs „kontextgebundene Regel".

Definition 8.3 (Kontextgebundene Regel) *Eine **kontextgebundene Ersetzungsregel** der Form $\vec{x} \rightarrow \vec{y}/\vec{z}_1\_\_\vec{z}_2$ erlaubt einen Übergang von $\vec{u}$ nach $\vec{v}$, falls es $\vec{w}_1$ und $\vec{w}_2$ gibt derart, dass $\vec{u} = \vec{w}_1\,\vec{z}_1\,\vec{x}\,\vec{z}_2\,\vec{w}_2$ und $\vec{v} = \vec{w}_1\,\vec{z}_1\,\vec{y}\,\vec{z}_2\,\vec{w}_2$.*

Satz 8.4 *Es sei $\rho = \vec{x} \to \vec{y}/\vec{z}_1\_\_\vec{z}_2$ und $\rho' = \vec{z}_1\vec{x}\vec{z}_2 \to \vec{z}_1\vec{y}\vec{z}_2$. Genau dann erlaubt ρ einen Übergang von $\vec{u}$ nach $\vec{v}$, wenn ρ' einen Übergang von $\vec{u}$ nach $\vec{v}$ erlaubt.*

Der Beweis ist einfach.

Auch wenn also kontextgebundene Regeln nichts anderes sind, als konventionelle Ersetzungsregeln, werden sie gerne verwendet, weil sie konzeptionell oft einfacher zu verstehen sind. Die Idee dahinter ist, dass es nicht $\vec{z}_1\vec{x}\vec{z}_2$ ist, welches ersetzt wird, sonderen nur $\vec{x}$ und dass $\vec{z}_1$ und $\vec{z}_2$ lediglich Bedingungen an das Vorkommen von $\vec{x}$ bilden. Diese Notation sollte man sich also merken, weil sie oft vorkommt. Der Kontext der Regel ist *nicht* notwendig das ganze Wort, in dem ersetzt wird. So erlaubt die Regel den Übergang von $\vec{w}_1\vec{z}_1\vec{x}\vec{z}_2\vec{w}_2$ nach $\vec{w}_1\vec{z}_2\vec{y}\vec{z}_2\vec{w}_2$, weil die Kontextbedingung erfüllt ist: Unmittelbar vor $\vec{x}$ befindet sich die Zeichenkette $\vec{z}_1$, und unmittelbar danach haben wir $\vec{z}_2$.

Kehren wir nun zur Pluralbindung im Englischen zurück. Der Plural wird bei Worten, die in /`s`/, /`y`/, /`ch`/ und /`sh`/ enden, anders gebildet. Endet das Wort in /`s`/, so wird /`ses`/ angehängt. Endet es in /`ch`/ oder /`sh`/, so wird /`es`/ angehängt. Endet es in /`y`/, so wird dies zu /`ies`/. (Wiederum übergehe ich jetzt einige weitere Fälle ebenso wie das Dutzend unregelmäßiger Nomina.) Jetzt haben wir folgende Regeln.

$$\begin{array}{lll} \varepsilon \to \texttt{ses} & / \, \texttt{s}\_\_\textvisiblespace \\ \varepsilon \to \texttt{es} & / \, \texttt{ch}\_\_\textvisiblespace \\ \varepsilon \to \texttt{es} & / \, \texttt{sh}\_\_\textvisiblespace \\ \texttt{y} \to \texttt{ies} & / \, \_\_\textvisiblespace \end{array} \tag{8.15}$$

Wir sagen nun, dass die Pluralbildung nicht eine einzige Regel ist sondern eine *Menge*. So etwas nennen wir ein Regelsystem.

Definition 8.5 (Regelsystem) *Ein **Regelsystem** ist eine Menge R von (kontextgebundenen) Ersetzungsregeln. Wir sagen, R **erlaube den Übergang von $\vec{u}$ nach $\vec{v}$**, in Zeichen $\vec{u} \to_R \vec{v}$, falls es eine Regel $\rho \in R$ gibt, die einen Übergang von $\vec{u}$ nach $\vec{v}$ erlaubt.*

Und wiederum definieren wir die Iteration der Übergangsrelation:

$$\vec{x} \to_R^0 \vec{y} :\Leftrightarrow \vec{x} = \vec{y} \tag{8.16}$$

$$\vec{x} \to_R^{n+1} \vec{y} :\Leftrightarrow \text{es existiert ein } \vec{z} \text{ mit } \vec{x} \to_R^n \vec{z} \to_R \vec{y} \tag{8.17}$$

$$\vec{x} \to_R^* \vec{y} :\Leftrightarrow \text{es existiert ein } n \in \mathbb{N} \text{ mit } \vec{x} \to_R^n \vec{y} \tag{8.18}$$

Kehren wir nun zur Pluralbildung zurück. Sei R das System der in (8.15) genannten Regeln. Dann haben wir

$$\begin{array}{l} \texttt{dog}\llcorner \to_R \texttt{dogs}\llcorner \\ \texttt{church}\llcorner \to_R \texttt{churches}\llcorner \\ \texttt{church}\llcorner \to_R \texttt{churchs}\llcorner \\ \texttt{fifty}\llcorner \to_R \texttt{fifties}\llcorner \\ \texttt{fifty}\llcorner \to_R \texttt{fiftys}\llcorner \end{array} \tag{8.19}$$

Wir erzeugen einige nicht korrekte Formen, weil wir die erste Regel nicht eingeschränkt haben. Sie darf auf einige Worte *nicht* anwendbar sein. Dies kann man erreichen, indem man die Regel in Unterfälle aufteilt, etwa so: $\varepsilon \to \texttt{s/a}\_\_\llcorner$, $\varepsilon \to \texttt{s/b}\_\_\llcorner$, und so weiter. Man hat dafür auch wieder Kurzschreibweisen, die einem erlauben, Kontexte zusammenzufassen, oder komplementäre Kontexte zu definieren, aber das will ich hier nicht weiter verfolgen. Es gibt allerdings meines Wissens kaum Regeln, die sich nicht als endliche Mengen von Regeln des in Definition 8.5 definierten Typs auffassen lassen.

Zu guter Letzt weise ich noch darauf hin, dass wir folgende Übergänge haben:

$$\texttt{dogs}\llcorner \to_R \texttt{dogsses}\llcorner, \quad \texttt{churches}\llcorner \to_R \texttt{churchesses}\llcorner \tag{8.20}$$

Abgesehen von der nicht ganz korrekten Bildung /`dogsses`␣/ ist es *kein* Fehler, dass die Pluralbildung auch auf Purale angewendet werden kann. Denn die Regeln sind nicht für die Frage zuständig, was für eine Form das ist. Zunächst einmal wird nämlich dasselbe Regelsystem auch angewendet für die Bildung der 3. Person Singular. Zweitens gibt es Worte, die formal wie Plurale aussehen, aber keine sind: /`yes`/, /`plus`/, und so weiter. Ob eine Form eine Pluralform ist oder nicht, kann man nicht allein am Aussehen ausmachen. Das ist eine Aufgabe der Morphologie (welche mehr umfasst als nur ein einfaches Regelsystem). Es gibt im Übrigen morphologische Prozesse, die durchaus mehrfach anwendbar sind.

Wir sind nun soweit, den Begriff einer Grammatik einzuführen. Diese fügt einem Regelsystem noch zwei Dinge hinzu: eine Bestimmung, wie eine Ableitung anfangen darf, und eine Bestimmung, wann sie beendet ist.

Definition 8.6 (Grammatik) *Eine **formale Grammatik** ist ein Quadrupel $G = \langle A, N, S, R\rangle$, wobei A das **terminale Alphabet**, N das **nichtterminale Alphabet**, $S \in N$ das **Startsymbol** und R ein endliches Regelsystem, d.h. eine endliche Menge von Ersetzungsregeln über $A \cup N$ ist. A und N seien endlich. Ferner wird verlangt, dass für jede Regel $\vec{x} \to \vec{y}/\vec{u}\_\_\vec{v} \in R$, $\vec{x}$ wenigstens ein nichtterminales Symbol enthält.*

Definition 8.7 (Sprache einer Grammatik) *Es sei $G = \langle A, N, S, R\rangle$ eine Grammatik. Wir schreiben $\vec{x} \vdash_G \vec{y}$, falls $\vec{x} \to_R^* \vec{y}$, das heißt, falls es ein $n \in \mathbb{N}$ gibt mit $\vec{x} \to_R^n \vec{y}$. Außerdem schreiben wir $\vdash_G \vec{y}$, wenn $S \vdash_G \vec{y}$. Schließlich ist*

$$L(G) = \{\vec{y} : \vdash_G \vec{y}\} \tag{8.21}$$

Es sei zum Beispiel G die folgende Grammatik: $G := \langle\{\mathtt{a}, \mathtt{b}\}, \{\mathtt{S}\}, \mathtt{S}, \{\mathtt{S} \to \mathtt{aSb}, \mathtt{S} \to \mathtt{ab}\}\rangle$. Dann ist

$$\begin{aligned} L(G) = & \ \{\mathtt{ab}, \mathtt{aabb}, \mathtt{aaabbb}, \cdots\} \\ = & \ \{\mathtt{a}^n\mathtt{b}^n : n \in \mathbb{N} - \{0\}\} \end{aligned} \tag{8.22}$$

Dies sehen wir wie folgt. Zu betrachten sind alle Ableitungen, die mit /S/ beginnen und in einer Zeichenkette über A enden. Dazu betrachten wir zunächst sämtliche Ableitungen, die mit /S/ beginnen. Ableitungen der Länge 1 gibt es zwei: $\langle\mathtt{S}, \mathtt{aSb}\rangle$, $\langle\mathtt{S}, \mathtt{ab}\rangle$. Die zweite Ableitung liefert uns schon das erste Wort in $L(G)$. Es sei angemerkt, dass man auf eine rein terminale Zeichenkette keine Ersetzung mehr anwenden kann, weil diese ja immer etwas ersetzen müssen, das mindestens ein Nichtterminalsymbol enthält. Also können wir jetzt die zweite Ableitung außen vorlassen, und uns um Fortsetzungen der ersten kümmern. Davon gibt es wieder zwei:

$$\langle\mathtt{S}, \mathtt{aSb}, \mathtt{aaSbb}\rangle, \langle\mathtt{S}, \mathtt{aSb}, \mathtt{aabb}\rangle \tag{8.23}$$

Die zweite Ableitung ergibt das nächste Wort, und so müssen wir uns nur die erste anschauen:

$$\langle\mathtt{S}, \mathtt{aSb}, \mathtt{aaSbb}, \mathtt{aaaSbbb}\rangle, \langle\mathtt{S}, \mathtt{aSb}, \mathtt{aaSbb}, \mathtt{aaabbb}\rangle \tag{8.24}$$

Und so weiter.

Ich weise hier auf eine besondere Notation hin. Wir schreiben die Regeln der obigen Grammatik auch einfacher

$$\mathtt{S} \to \mathtt{ab} \mid \mathtt{aSb} \tag{8.25}$$

Hierbei zeigt der vertikale Strich '|' an, dass wir zwischen der linken und der rechten Seite wählen können. Die „Regel" (8.25) ist also in Wirklichkeit zwei Regeln.

Hier ist ein ähnliches Beispiel, welches ich von Steven Pinker aus dem Buch „The Language Instinct" übernommen habe. Die Grammatik des Wettrüstens ist

$$\mathtt{S} \to \ \mathtt{Anti\text{-}Sn\text{-}Rakete} \mid \mathtt{Rakete} \tag{8.26}$$

Hierbei sind /A/ und /R/ Buchstaben, während /S/ ein Nichtterminalzeichen ist. Dies ist bewusst so organisiert, damit das Beispiel realistisch ist. In der Notation, die weiter unten vorgeschlagen wird, würden wir anstelle von /S/ <S> schreiben, was die Regeln etwas klarer machen würde.

Daraus kann man unter anderem die folgenden Worte stricken:

(8.27)

```
Rakete
Anti-Raketen-Rakete
Anti-Anti-Raketen-Raketen-Rakete
```

Nehmen wir noch eine Grammatik. Es sei A unser normales Alphabet der Kleinbuchstaben. Ferner sei $N = \{\mathtt{S}, \mathtt{D}, \mathtt{U}, \mathtt{R}, \mathtt{P}, \mathtt{N}, \mathtt{V}, \mathtt{I}\}$, das Startsymbol sei wieder S, und R enthalte die folgenden Regeln:

(8.28)

S→UP	U→DN | DNR
D→the␣ | a␣	N→car␣ | rat␣ | mouse␣
R→TP	T→that␣
P→I | VU	I→ran␣ | drove␣
V→drove␣ | chased␣	

Diese Grammatik erzeugt sinnvolle englische Sätze. Hier ist eine vollständige Ableitung eines Satzes:

(8.29)

```
S
UP
DNRP
the␣NRP
the␣rat␣RP
the␣rat␣TPP
the␣rat␣that␣PP
the␣rat␣that␣VUP
the␣rat␣that␣drove␣UP
the␣rat␣that␣drove␣DNP
the␣rat␣that␣drove␣a␣NP
the␣rat␣that␣drove␣a␣car␣P
the␣rat␣that␣drove␣a␣car␣I
the␣rat␣that␣drove␣a␣car␣ran␣
```

An dieser Stelle sei angemerkt, dass dieser Satz auch noch unzählige andere Ableitungen hat. Zum Beispiel hätten wir anstelle des Übergangs von /UP/ zu /DNRP/

auch erst einmal /UP/ in /Uran␣/ überführen können, um dann im zweiten Schritt /DNran␣/ zu gewinnen.

Da unser normales Alphabet relativ klein ist, behilft man sich mit ein paar Hilfsnotationen, um einen beliebig großen Vorrat an Nichtterminalzeichen zu bekommen. Eine Möglichkeit, die gerne in der Informatik gebraucht wird, ist, aus eine beliebigen Zeichenkette mittels spitzer Klammern ein einziges Symbol zu machen. Zum Beispiel ist <NP> ein einziges Symbol, und das deswegen auch von der Zeichenkette /NP/ sowie den anderen Symbolen verschieden ist. Es ist also <NP> verschieden von <N>, und von <VP>. Diese Notation ist nützlich, wenn das eigentliche Alphabet sowieso komplett vergeben ist (als Terminalalphabet), denn dann können wir das Nichtterminalzeichen <N> von dem Terminalzeichen N unterscheiden. Wir können zum Beispiel folgende Regel aufschreiben:

(8.30) <S> → <NP><VP>

Diese besagt, dass das Nichtterminalzeichen <S> durch das Wort der Länge 2 bestehend aus <NP> gefolgt von <VP> ersetzt werden kann. Ebenso können wir die Regel (8.26) wie folgt notieren.

(8.31) <S> → Anti-<S>n-Rakete | Rakete

Diese Notation ist recht konzis (was natürlich daran liegt, dass die spitzen Klammern recht selten verwendet werden und deshalb den speziellen Charakter der Symbole sofort zeigen).

Eine andere, weniger robuste Notation wird in der Linguistik verwendet. Hier werden die Nichtterminalzeichen von den Terminalzeichen dadurch unterschieden, dass sie mit einem Großbuchstaben beginnen. Nichtterminalzeichen haben beliebe Länge. Als Trennsymbol zwischen Nichtterminalsymbolen und anderen Zeichen dient das Leerzeichen. (8.30) liest sich dann so

(8.32) S → NP␣VP

Hierbei ist ␣ die sinnliche Hervorhebung des Leerzeichens; normalerweise wird es natürlich gar nicht notiert, also sieht man eigentlich nur dieses:

(8.33) S → NP VP

Dies ist zwar recht kurz, hat aber den Nachteil, dass man keine großen Grammatiken damit schreiben kann, weil man bei steigender Anzahl von Nichtterminalzeichen unweigerlich in Notationskonflikte kommt.

Übung

[33] Es sei $\rho = \langle \vec{x}, \vec{y} \rangle$ eine Ersetzungsregel. Dann ist $\rho^{\smile} := \langle \vec{y}, \vec{x} \rangle$ ebenfalls eine Ersetzungsregel. Für ein Regelsystem R ist $R^{\smile} := \{\rho^{\smile} : \rho \in R\}$ ein Regelsystem. Zeigen Sie, dass für zwei beliebige Zeichenketten $\vec{u}$ und $\vec{v}$ Folgendes gilt.

1. Genau dann erlaubt ρ einen Übergang von $\vec{u}$ nach $\vec{v}$, wenn $\rho^{\smile}$ einen Übergang von $\vec{v}$ nach $\vec{u}$ erlaubt.

2. Genau dann erlaubt R einen Übergang von $\vec{u}$ nach $\vec{v}$, wenn $R^{\smile}$ einen Übergang von $\vec{v}$ nach $\vec{u}$ erlaubt.

3. $\vec{u} \to_R^n \vec{v}$ genau dann, wenn $\vec{v} \to_{R^{\smile}}^n \vec{u}$. *Hinweis.* Induktion.

4. $\vec{u} \to_R^* \vec{v}$ genau dann, wenn $\vec{v} \to_{R^{\smile}}^* \vec{u}$.

[34] Die Formen des schwachen Verbs im Deutschen sind wie folgt: Infinitiv /`loben`/, /`lachen`/, /`sagen`/, Partizip Perfekt /`gelobt`/, /`gelacht`/, /`gesagt`/.

1. Fomulieren Sie eine Regel für die Bildung des Infinitivs aus dem Verbstamm.

2. Formulieren Sie zwei Regeln ρ_1 und ρ_2 derart, dass das Partizip Perfekt aus dem Stamm durch je einmaliges Anwenden von ρ_1 und dann ρ_2 hervorgeht.

Überlegen Sie: kann man die Regeln auf das Ergebnis erneut anwenden? (Natürlich nur formal gesehen, nicht im Deutschen!) Gibt es Formen, die so aussehen, als seien sie das Ergebnis der Regeln, aber sind es im Deutschen gar nicht? Kann man auch erst ρ_2 anwenden und dann ρ_1?

[35]

1. Überführen Sie die Regeln `aabc` $\to$ `abcc` sowie `abc` $\to$ `ac` jeweils in eine kontextgebundene Regel mit echter Kontextbedingung.

2. Machen Sie aus der Regel `ab` $\to$ `a`$/\varepsilon$__`c` eine Ersetzungsregel ohne Kontextbedingung.

[36] Sei `aa` $\to$ `ba` eine Ersetzungsregel. Wie viele verschiedene Zeichenketten bekommen wir durch einmaliges Anwenden auf /`aaaa`/ bzw. /`caca`/?

[37] Es sei $\vec{x} \to \vec{y}$ eine Ersetzungsregel und $\vec{z}$ eine Zeichenkette. Wie viele verschiedene Zeichenketten kann man aus $\vec{z}$ höchstens durch einmalige Anwendung dieser Regel bekommen? *Anleitung.* Wir unterscheiden zwei Fälle. Der erste Fall ist, dass $\vec{x}$ in $\vec{z}$ nicht vorkommt. In diesem Fall gibt es kein Ergebnis. Nun nehmen wir an, $\vec{x}$ komme in $\vec{z}$ vor. Geben Sie in Abhängigkeit von der Länge von $\vec{x}$ und $\vec{z}$ an, wie viele Vorkommen von $\vec{x}$ es höchstens geben kann.

[38] In den nun folgenden Übungen betrachten wir Regeln, welche ein geschriebenes Wort in eine Phonemkette übersetzen. Das Alphabet bestehe aus der Vereinigung von dem lateinischen Alphabet (mit Groß- und Kleinschreibung), sowie einigen Hilfszeichen. Einfache Fälle sind `p` → p und `P` → p. Ebenso `o` → o und `l` → l. Dreifache Anwendung dieser Regeln macht aus /`Pol`/ die Sequenz [pol]. (Schrägstriche zitieren geschriebene Worte, eckige Klammern bezeichnen Phonemsequenzen.) Schreiben Sie eine Ableitung auf.

[39] Allerdings geht es mit /`b`/ schon los: das Ende einer Silbe (genauer: die Koda) wird stimmlos. Wir müssen also zwei Regeln formulieren: `b` → b und `b` → p. Führen Sie ein Symbol für die Silbengrenze ein (etwa ı) und formulieren Sie nun die Regeln für die Übersetzung von /`b`/. *Anmerkung.* Die Silbengrenze wird jetzt im geschriebenen Wort markiert (eigentlich denkt die Linguistik genau umgekehrt: man setzt Phoneme in Buchstaben um). Die Koda besteht aus mehreren Konsonanten. Man müsste also die Regeln mehrfach anwenden. Das bedeutet aber, dass manche Konsonanten nicht an der Silbengrenze stehen, etwa in /`Halt`/. In diesem Fall müssen wir uns daran erinnern, dass Okklusive in der Regel final in der Koda sein müssen (also gibt es nicht /`tl`/ in der Koda; jedoch besehe man sich das Wort /`Herbst`/). Glücklicherweise ist bei nicht Okklusiven der Kontrast stimmlos/stimmhaft nicht phonematisch.

[40] Spezielle Vorkehrungen müssen wir für die Kombinationen /`ch`/ und /`sch`/. Erster Versuch: `ch` → x, `ch` → ç und `sch` → ʃ. Erkennen Sie, welche Probleme wir jetzt mit dem Wort /`bisschen`/ bekommen? Gibt es eine Lösung?

[41] Man formuliert kontextgebundene Regeln auch mit Mengen von Kontexten, etwa so: `b` → `b` / V__. Dies bedeutet soviel wie: /`b`/ wird zu [b], sofern links davon ein Vokal steht. Dies entspricht einer endlichen Menge von kontextsensitiven Regeln im Standardformat. Welche sind dies?

Kapitel 9

Reguläre Sprachen

Sprachen sind Mengen von Zeichenketten. Sie lassen sich folglich wie Mengen manipulieren. Mit zwei Sprachen L und M sind auch die Vereinigung $L \cup M$, der Schnitt $L \cap M$ und das relative Komplement $L - M$ Sprachen. Von sehr wichtiger technischer Bedeutung ist jedoch die **Verkettung** oder das **Produkt** $L \cdot M$.

Definition 9.1 (Verkettung von Sprachen) *Es seien L und M Sprachen über A. Dann definieren wir*

$$L \cdot M := \{\vec{x} \cdot \vec{y} : \vec{x} \in L \textit{ und } \vec{y} \in M\} \tag{9.1}$$

Wir schreiben auch LM anstelle von $L \cdot M$. Man beachte, dass $L \cdot M$ aus allen Verkettungen $\vec{x} \cdot \vec{y}$ besteht, wo $\vec{x} \in L$ und $\vec{y} \in M$. Deswegen ist $\{\mathsf{a}, \mathsf{ab}\} \cdot \{\mathsf{d}, \mathsf{da}\} = \{\mathsf{ad}, \mathsf{ada}, \mathsf{abd}, \mathsf{abda}\}$. Ebenso ist $\{\mathsf{a}, \mathsf{ab}\} \cdot \{\mathsf{a}, \mathsf{ab}\} = \{\mathsf{aa}, \mathsf{aab}, \mathsf{aba}, \mathsf{abab}\}$. Das Multiplikationszeichen quotedblase·” bindet stärker als die Mengenoperationen, insbesondere die Vereinigung von Mengen. Deswegen könnte ich in der Formulierung des folgenden Lemmas einige Klammern sparen (und zwar in (d) und (e) auf der rechten Seite des Gleichheitszeichens, nicht jedoch auf der linken).

Lemma 9.2 *Es gelten folgende Gesetze.*

$$\begin{array}{lll} (a) & \{\varepsilon\} \cdot M & = M \\ (b) & M \cdot \{\varepsilon\} & = M \\ (c) & L \cdot (M \cdot N) & = (L \cdot M) \cdot N \\ (d) & L \cdot (M \cup N) & = (L \cdot M) \cup (L \cdot N) \\ (e) & (M \cup N) \cdot L & = (M \cdot L) \cup (N \cdot L) \end{array} \tag{9.2}$$

Beweis. Das sieht man so. (a) $\{\varepsilon\} \cdot M = \{\varepsilon \cdot \vec{x} : \vec{x} \in M\} = \{\vec{x} : \vec{x} \in M\} = M$. Ebenso für (b). Zu (c).

$$
\begin{aligned}
L \cdot (M \cdot N) &= \{\vec{x} \cdot \vec{u} : \vec{x} \in L, \vec{u} \in M \cdot N\} \\
&= \{\vec{x} \cdot \vec{u} : \vec{x} \in L, \vec{u} \in \{\vec{y} \cdot \vec{z} : \vec{y} \in M, \vec{z} \in N\}\} \\
&= \{\vec{x} \cdot (\vec{y} \cdot \vec{z}) : \vec{x} \in L, \vec{y} \in M, \vec{z} \in N\} \\
&= \{(\vec{x} \cdot \vec{y}) \cdot \vec{z} : \vec{x} \in L, \vec{y} \in M, \vec{z} \in N\} \\
&= \{\vec{v} \cdot \vec{z} : \vec{v} \in L \cdot M, \vec{z} \in N\} \\
&= (L \cdot M) \cdot N
\end{aligned}
\tag{9.3}
$$

Zu (d).

$$
\begin{aligned}
L \cdot (M \cup N) &= \{\vec{x} \cdot \vec{u} : \vec{x} \in L, \vec{u} \in M \cup N\} \\
&= \{\vec{x} \cdot \vec{u} : [\vec{x} \in L \text{ und } \vec{u} \in M] \text{ oder } [\vec{x} \in L \text{ und } \vec{u} \in N]\} \\
&= \{\vec{x} \cdot \vec{u} : \vec{x} \in L, \vec{u} \in M\} \cup \{\vec{x} \cdot \vec{u} : \vec{x} \in L, \vec{u} \in N\} \\
&= (L \cdot M) \cup (L \cdot N)
\end{aligned}
\tag{9.4}
$$

Ebenso (e). ⊣

Es ist aber im Allgemeinen $L \cdot M \neq M \cdot L$. Dazu überlege man sich, dass ja auch $\vec{x} \cdot \vec{y} \neq \vec{y} \cdot \vec{x}$ ist — zum Beispiel $\mathsf{a} \cdot \mathsf{b} \neq \mathsf{b} \cdot \mathsf{a}$ —, sodass also $\{\mathsf{a}\} \cdot \{\mathsf{b}\} = \{\mathsf{ab}\} \neq \{\mathsf{ba}\} = \{\mathsf{b}\} \cdot \{\mathsf{a}\}$.

Beim Produkt von Sprachen verhält sich $\{\varepsilon\}$ also genauso, wie ε bei der Verkettung von Zeichenketten oder Wörtern. Es sei angemerkt, dass bei der Notation das Produkt stärker bindet als die Vereinigung. Es ist also $L \cdot M \cup N$ unmißverständlich: Es ist zu lesen als $(L \cdot M) \cup N$ und nicht als $L \cdot (M \cup N)$.

Es ist üblich, folgende Abkürzungen zu benutzen:

$$\vec{x}^{\,0} := \varepsilon, \qquad \vec{x}^{\,n+1} := \vec{x}^{\,n} \cdot \vec{x} \tag{9.5}$$

Diese Notation benutzen wir auch für Sprachen. Es ist dann

$$L^0 := \{\varepsilon\}, \qquad L^{n+1} := L^n \cdot L \tag{9.6}$$

Besonders wichtig ist der *Kleene-Stern* $(-)^*$.

Definition 9.3 (Kleene Stern) *Es sei L eine Sprache. Dann bezeichnet L^* die Sprache*

$$L^* := \bigcup_{n \in \mathbb{N}} L^n = L^0 \cup L^1 \cup L^2 \cup \cdots \tag{9.7}$$

Es ist also L^* die Menge aller endlichen Wiederholungen von L.

Der Stern bindet stärker als „·" und die Mengenoperationen. Ich weise gleich darauf hin, dass im Allgemeinen *nicht* $L^n = \{\vec{x}^n : n \in \mathbb{N}\}$. Ein Beispiel dazu ist $L := \{\mathsf{a}, \mathsf{b}\}$. Dann haben wir

$$L \cdot L = \{\mathsf{a}, \mathsf{b}\} \cdot \{\mathsf{a}, \mathsf{b}\} = \{\mathsf{aa}, \mathsf{ab}, \mathsf{ba}, \mathsf{bb}\} \neq \{\mathsf{aa}, \mathsf{bb}\} \tag{9.8}$$

Der Kleene-Stern ist deswegen so bedeutsam, weil die Sprache L^* eine einfache Rekursionsgleichung löst. Ich bringe zunächst ein Beispiel. Es sei folgende Gleichung gegeben.

$$X = \{\mathsf{ab}\} \cdot X \cup \{\mathsf{a}, \mathsf{ca}\} \tag{9.9}$$

Wir wollen diejenigen Mengen X bestimmen, die diese Gleichung erfüllen. Aufgrund der Gleichung ist $M := \{\mathsf{a}, \mathsf{ca}\} \subseteq X$. Wir setzen also $X_0 := \{\mathsf{a}, \mathsf{ca}\}$. Es ist dann $X_0 \subseteq X$. Da nun auch $\{\mathsf{ab}\} \cdot X \subseteq X$, so ist auch $\{\mathsf{ab}\} \cdot X_0 \subseteq X$, also sind aba und abca auch in X. Wir setzen also

$$X_1 := \{\mathsf{ab}\} \cdot X_0 \cup M = \{\mathsf{a}, \mathsf{ca}, \mathsf{aba}, \mathsf{abca}\} \tag{9.10}$$

Es ist also $X_1 \subseteq X$. Und dann setzen wir

$$X_2 := \{\mathsf{ab}\} \cdot X_1 \cup M = \{\mathsf{a}, \mathsf{ca}, \mathsf{aba}, \mathsf{abca}, \mathsf{ababa}, \mathsf{ababca}\} \tag{9.11}$$

Und schließlich setzen wir ganz allgemein

$$X_{n+1} := \{\mathsf{ab}\} \cdot X_n \cup M \tag{9.12}$$

Wir haben dann

$$X_0 \subseteq X_1 \subseteq X_2 \subseteq \cdots \tag{9.13}$$

Ich stelle hier ohne weitere Rechnung fest, dass

$$X_n = \left(\bigcup_{i \le n} \{\mathsf{ab}\}^i\right) \cdot \{\mathsf{a}, \mathsf{ca}\} = \{(\mathsf{ab})^i\mathsf{a}, (\mathsf{ab})^i\mathsf{ca} : i \le n\} \tag{9.14}$$

Die gesuchte Menge ist jetzt die kleinste Menge, die alle X_n enthält; dies ist die Vereinigung aller der X_n, und es ist leicht zu sehen, dass

$$X = \{(\mathsf{ab})^n\mathsf{a}, (\mathsf{ab})^n\mathsf{ca} : n \in \mathbb{N}\} = \{\mathsf{ab}\}^* \cdot \{\mathsf{a}, \mathsf{ca}\} \tag{9.15}$$

eine Lösung ist.

Proposition 9.4 *Es ist $L^* \cdot M$ die kleinste Sprache X, für die gilt*

$$X = L \cdot X \cup M \tag{9.16}$$

Ist darüberhinaus $\varepsilon \notin L$, so ist $L^* \cdot M$ auch die einzige Lösung von (9.16).

Beweis. Wir zeigen als Erstes, dass $X \supseteq L^n \cdot M$ für alle $n \in \mathbb{N}$. Daraus folgt zunächst, dass $X \supseteq L^* \cdot M$ sein muss. Beginnen wir mit $n = 0$. Es muss sicher $X \supseteq M$ gelten, aufgrund von (9.16). Da $L^0 \cdot M = \{\varepsilon\} \cdot M = M$, ist das die Behauptung für $n = 0$. Nun zeigen wir, dass die Behauptung für $n + 1$ gilt, sofern sie für n gilt (Induktionsbeweis). Angenommen, $X \supseteq L^n \cdot M$ sei gezeigt. Sei $\vec{x} \in L^{n+1} \cdot M$. Dann ist $\vec{x} = \vec{u}\,\vec{v}$ mit $\vec{u} \in L$ und $\vec{v} \in L^n \cdot M$. Dann ist nach Induktionsvoraussetzung $\vec{v} \in X$, und wegen (9.16) auch $\vec{x} \in X$. Dies zeigt die Behauptung.

Daraus folgt unmittelbar, dass für jede Lösung der Gleichung X gelten muss $X \supseteq L^* \cdot M$.

Nun zeigen wir noch, dass die Sprache $L^* \cdot M$ die Gleichung (9.16) erfüllt.

$$\begin{aligned}
(\textstyle\bigcup_{n\in\mathbb{N}} L^n) \cdot M \quad &= \textstyle\bigcup_{n\in\mathbb{N}} L^n \cdot M \\
&= (\textstyle\bigcup_{n>0} L^n \cdot M) \cup L^0 \cdot M \\
&= L \cdot (\textstyle\bigcup_{n\in\mathbb{N}} L^n \cdot M) \cup M \\
&= L \cdot (L^* \cdot M) \cup M
\end{aligned} \tag{9.17}$$

Damit ist $L^* \cdot M$ die kleinste Sprache, die die Gleichung (9.16) erfüllt.

Sei nun $\varepsilon \notin L$ und sei X irgendeine Lösung der Gleichung. Wir wollen zeigen, dass $X = L^* \cdot M$. Wir nehmen an, dass dies nicht so ist. Dann ist aber, wie schon gesehen, $L^* \cdot M \subsetneq X$. Sei also $\vec{x}$ eine Zeichenkette kleinster Länge aus X, für die $\vec{x} \notin L^* \cdot M$. Dann ist $\vec{x} \notin M$. Aufgrund der Gleichung (9.16) ist dann $\vec{x} \in L \cdot X$, und so gibt es $\vec{y} \in L$ und $\vec{u} \in X$ mit $\vec{x} = \vec{y}\,\vec{u}$. Nach Annahme über L ist $\vec{y} \neq \varepsilon$ und so ist $|\vec{u}| < |\vec{x}|$. Da $\vec{x}$ minimal war, ist $\vec{y} \in L^* \cdot M$, also $\vec{y} \in L^n \cdot M$ für ein gewisses n. Dann ist aber $\vec{x} \in L^{n+1} \cdot M$, also $\vec{x} \in L^* \cdot M$, im Widerspruch zur Annahme. ⊣

Der Fall $\varepsilon \in L$ ist ganz anders. Sehen wir ihn uns näher an. Sei K eine beliebige Menge mit $M \subseteq K \subseteq A^*$. Setze nun $X_0 := K$ und rekursiv

$$X_{n+1} := L \cdot X_n \cup M \tag{9.18}$$

Dann gilt $X_n \subseteq X_{n+1}$, weil nämlich $\varepsilon \in L$. Und so haben wir eine unendlich aufsteigende Kette:

$$X_0 \subseteq X_1 \subseteq X_2 \subseteq \cdots \tag{9.19}$$

(Wobei durchaus sein kann, dass für ein n gilt $X_{n+1} = X_n$. In diesem Fall ist dann $X_{n+k} = X_n$ für alle k, und die Kette besteht nur aus endlich vielen *verschiedenen* Mengen.) Es ist nicht schwer zu sehen, dass die kleinste Menge, die all diese Mengen enthält, genau die Vereinigung ist:

$$X_* := \bigcup_{n\in\mathbb{N}} X_n \tag{9.20}$$

Diese erfüllt die Gleichung. Denn es ist

$$\begin{aligned} L \cdot X_* \cup M &= L \cdot (\textstyle\bigcup_{n\in\mathbb{N}} X_n) \cup M \\ &= (\textstyle\bigcup_{n\in\mathbb{N}} L \cdot X_n) \cup M \\ &= \textstyle\bigcup_{n\in\mathbb{N}} (L \cdot X_n \cup M) \\ &= \textstyle\bigcup_{n\in\mathbb{N}} X_{n+1} \\ &= \textstyle\bigcup_{n\in\mathbb{N}} X_n \\ &= X_* \end{aligned} \tag{9.21}$$

Die letzte Gleichung gilt wiederum, weil $X_0 \subseteq X_1$.

Es ist ferner ganz allgemein für $\varepsilon \notin L$

$$X_n = L^n \cdot M. \tag{9.22}$$

Falls $\varepsilon \in L$, so ist

$$X_n = L^n \cdot K. \tag{9.23}$$

Obwohl (9.22) ein Spezialfall von (9.23) ist, zeigen wir nur das Erste. Dies ist sicher richtig für $n = 0$. Nun führen wir eine Induktion durch. Dazu nehmen wir an, $X_n = L^n \cdot M$. Dann ist $X_{n+1} = L \cdot X_n \cup M = L \cdot L^n \cdot M \cup M = L^{n+1} \cdot M \cup M = L^{n+1} \cdot M$, weil $M \subseteq L^{n+1} \cdot M$ ist, denn $\varepsilon \in L^{n+1}$. Zu guter Letzt folgt jetzt leicht, dass

$$X_* = L^* \cdot M \tag{9.24}$$

Ich weise noch auf Folgendes hin. Ist $L = \{\varepsilon\}$, so ist, wie man leicht sieht, $X_1 = X_2$. Denn $L^n = \{\varepsilon\}$, und so ist $X_n = L^n \cdot K$. Die Bedingung $X = L \cdot X \cup M$ reduziert sich auf $X = X \cup M$, was nichts anderes bedeutet, als dass $X \supseteq M$. Ist $L \neq \{\varepsilon\}$, so enthält L wenigstens ein Wort der Länge > 0, und so ist eine Lösung von $X = L \cdot X \cup M$ stets unendlich.

Wir kommen nun zu den regulären Sprachen.

Definition 9.5 (Regulärer Ausdruck) *Ein **regulärer Ausdruck über** A hat die folgende Form.*

① 0*, ε oder a, wo $a \in A$;*

② *s^*, wo s ein regulärer Ausdruck ist;*

③ *$s \cup t$, $s \cdot t$, wo s und t reguläre Ausdrücke sind.*

Eine andere Notation zu $\cup$ ist auch $|$. Außerdem wird der Punkt $\cdot$ gerne weggelassen. Terme werden sparsam geklammert. Der Punkt und die Vereinigung sind assoziativ. So schreibt man den Term $((\mathtt{M} \cdot \mathtt{a}) \cdot \mathtt{u}) \cdot \mathtt{s}$ unter Weglassen von Klammern schlicht $\mathtt{M} \cdot \mathtt{a} \cdot \mathtt{u} \cdot \mathtt{s}$; und schließlich lässt man auch noch die Punkte weg und bekommt `Maus`. Dies suggeriert natürlich, dass wir die Zeichenkette /`Maus`/ definiert haben, und das ist in der Tat auch richtig, wenn wir von der Tatsache absehen, dass es nicht diese *Zeichenkette* definiert, sondern die *Sprache* {`Maus`}, die als einziges Element ebendiese Zeichenkette enthält.

Doch zunächst muss ich sagen, welche Sprache ein regulärer Term eigentlich bezeichnet. Die von dem regulären Ausdruck r bezeichnete Sprache $S(r)$ ist wie folgt definiert.

$$\begin{aligned} S(0) &:= \varnothing \\ S(\varepsilon) &:= \{\varepsilon\} \\ S(a) &:= \{a\} \\ S(s \cup t) &:= S(s) \cup S(t) \\ S(s \cdot t) &:= S(s) \cdot S(t) \\ S(s^*) &:= S(s)^* \end{aligned} \tag{9.25}$$

Damit haben wir zum Beispiel

$$\begin{aligned} S(\mathsf{a}(\mathsf{b} \mid \mathsf{c})^*(\mathsf{ac} \mid \mathsf{a})) &= S(\mathsf{a})(S(\mathsf{b} \mid \mathsf{c}))^* S(\mathsf{ac} \mid \mathsf{a}) \\ &= \{\mathsf{a}\}(S(\mathsf{b} \mid \mathsf{c})^*)(S(\mathsf{ac}) \cup S(\mathsf{a})) \\ &= \{\mathsf{a}\}((S(\mathsf{b}) \cup S(\mathsf{c}))^*)(\{\mathsf{ac}\} \cup \{\mathsf{a}\}) \\ &= \{\mathsf{a}\}\{\mathsf{b}, \mathsf{c}\}^*\{\mathsf{ac}, \mathsf{a}\} \\ &= \{\mathsf{a}\}\{\varepsilon, \mathsf{b}, \mathsf{c}, \mathsf{bb}, \mathsf{bc}, \mathsf{cb}, \mathsf{cc}, \cdots\}\{\mathsf{ac}, \mathsf{a}\} \\ &= \{\mathsf{a}, \mathsf{ab}, \mathsf{ac}, \mathsf{abb}, \mathsf{abc}, \mathsf{acb}, \mathsf{acc}, \cdots\}\{\mathsf{ac}, \mathsf{a}\} \\ &= \{\mathsf{aac}, \mathsf{abac}, \mathsf{acac}, \mathsf{abbac}, \mathsf{abcac}, \\ &\qquad \mathsf{acbac}, \mathsf{accac}, \cdots, \\ &\quad\ \mathsf{aa}, \mathsf{aba}, \mathsf{aca}, \mathsf{abbac}, \mathsf{abca}, \mathsf{acba}, \\ &\qquad \mathsf{acca}, \cdots\} \end{aligned} \tag{9.26}$$

Definition 9.6 (Reguläre Sprache) *Eine Sprache L ist **regulär**, falls ein regulärer Term r existiert mit $L = S(r)$.*

Satz 9.7 *Die folgenden sind reguläre Sprachen:*

① *Die Ein-Wort Sprache $\{\vec{x}\}$.*

② *$L \cup M$, wenn L und M regulär sind.*

③ *Jede endliche Sprache.*

④ *$A^* - L$, wenn L regulär ist.*

⑤ *$L \cap M$, wenn L und M regulär sind.*

Daraus folgt, dass zum Beispiel auch die Sprache $A^* - \{\vec{x}\}$ („alles außer $\vec{x}$") auch regulär ist. Wir beginnen mit der ersten Behauptung, ①. Diese beweisen wir durch Induktion über die Länge n von $\vec{x}$. Ist $n = 0$, so ist $\{\vec{x}\} = \{\varepsilon\} = S(\varepsilon)$, also regulär. Ist $n > 0$, so ist $\vec{x} = a \cdot \vec{y}$, also $\{\vec{x}\} = \{a\} \cdot \{\vec{y}\}$. Nach Induktionsannahme ist $\{\vec{y}\} = S(r)$ für einen regulären Ausdruck r. Dann ist $\{\vec{x}\} = S(a \cdot r)$.

② ist leicht. Ist $L = S(r)$ und $M = S(s)$, so ist $L \cup M = S(r \cup s)$. Nun zu ③. Auch dies zeigen wir durch Induktion, diesmal über die Mächtigkeit n von L. Ist $n = 0$, so ist $L = \varnothing = S(0)$, also regulär. Ist $n > 0$, so ist $L = \{\vec{x}\} \cup M$, wobei $|M| = n - 1$. Nach Induktionsannahme ist $M = S(r)$ für ein reguläres r; nach ① ist $\{\vec{x}\} = S(s)$ für ein reguläres s. Dann ist $L = S(r \cup s)$.

Den Beweis der vierten Behauptung werde ich später (genauer im nächsten Kapitel) erbringen. Die fünfte folgt aus der vierten so: es sei A genügend groß, sodass $L, M \subseteq A^*$. Dann gilt $L \cap M = A^* - ((A^* - L) \cup (A^* - M))$. Reguläre Sprachen sind, wie wir gerade gesehen haben, unter Vereinigung und Komplement abgeschlossen.

Zum Schluss gebe ich noch ein Beispiel einer Sprache, die nicht regulär ist. Dies ist die Sprache $\{\varepsilon, \mathtt{ab}, \mathtt{aabb}, \mathtt{aaabbb}, \cdots\} = \{\mathtt{a}^n\mathtt{b}^n : n \in \mathbb{N}\}$. Diese Sprache ist nicht endlich, und ihr Komplement in $\{\mathtt{a}, \mathtt{b}\}^*$ ist auch nicht endlich (zum Beispiel gehören alle Zeichenketten der Form $\mathtt{a}^n\mathtt{b}^{n+1}$ dazu). Das genügt natürlich nicht für einen strengen Beweis, aber jedenfalls fällt die Sprache nicht in die Liste von Satz 9.7. Eine andere nichtreguläre Sprache ist die von der Grammatik (8.26). Diese ist

$$\{(\mathtt{Anti-})^n(\mathtt{Raketen-})^n\mathtt{Rakete} : n \in \mathbb{N}\} \tag{9.27}$$

Übungen

[42] Eine Menge L von Zeichenketten heißt eine Sprache, wenn sie eine Sprache über einem endlichen Alphabet ist. Zeigen Sie, dass mit L and M auch $L \cup M$, $L \cap M$ und $L - M$ Sprachen sind.

[43] Es sei $A = \{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d}\}$. Sei $L = \{\mathsf{a}, \mathsf{bc}, \mathsf{dd}\}$ und $M = \{\mathsf{cc}, \mathsf{b}\}$. Bechnen Sie $L \cdot L$, $L \cdot M$ und $M \cdot L$.

[44] Es sei $A = \{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d}\}$. Schreiben Sie einen regulären Ausdruck, dessen Sprache alle Zeichenketten enthält, die entweder mit c beginnen oder mit einer geraden Zahl von d aufhören.

[45] Die symmetrische Differenz von zwei Mengen M, N ist definiert durch $M\Delta N := (M - N) \cup (N - M)$. Zeigen Sie: ist M and N regulär, so auch $M\Delta N$.

[46] Es sei $A = \{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d}\}$. Schreiben Sie alle Zeichenketten auf, die unter $(\mathsf{ab}^* \mid \mathsf{d})(\mathsf{c} \mid \mathsf{d})^*$ fallen und die Länge höchsten 4 haben.

Kapitel 10

Endliche Automaten

Ein endlicher Automat ist eine Maschine, die eine Zeichenkette einliest und dann-entscheidet, ob sie angenommen wird oder nicht. Abstrakt definiert der Automat also eine Funktion, die zu jeder Zeichenkette entweder „ja” oder „nein” sagt. Der Automat liest die Zeichenkette von links nach rechts ein, Buchstabe für Buchstabe. Ist ein Buchstabe gelesen, schaut der Automat anschließend nicht mehr darauf. Ein Gedächtnis hat der Automat nur in Form von endlich vielen sogenannten Zuständen.

Definition 10.1 (Endlicher Automat) *Ein* ***endlicher Automat*** *ist ein Quintupel* $\mathfrak{A} = \langle A, Q, i, F, \delta\rangle$. *Hierbei ist*

- *A eine endliche Menge, das* ***Alphabet****,*
- *Q eine endliche Menge, die Menge der* ***inneren Zustände****,*
- *$i \in Q$ der* ***initiale Zustand****,*
- *$F \subseteq Q$ die Menge der* ***finalen Zustände*** *und*
- *$\delta \subseteq Q \times A \times Q$ die* ***Übergangsrelation****.*

Man überlegt sich leicht, dass auch δ eine endliche Menge ist.

Hier ist ein Beispiel. Der Automat $\mathfrak{B}$ akzeptiere alle Münzen ab 5 Cent. Der Nutzer darf so viel Geld einwerfen, wie er will. Falls der Nutzer exakt 25 Cent einwirft, bekommt er eine Ware. Überzahlt er, ohne vorher auf 25 Cent gekommen zu sein, so springt der Automat auf q_0 (den Startzustand). Eine nicht akzeptierte Münze wird übergangen, ohne dass sich der Zustand ändert. Der Einfachheit halber ignorieren wir diese Münzen.

(10.1) $A := \{5, 10, 20, 50, 100, 200\}$

Die Zustände benennen wir nach der Menge des gutgeschriebenen Geldes (nicht des eingeworfenen, weil ja überzahltes Geld einbehalten wird).

(10.2) $Q := \{q_0, q_5, q_{10}, q_{15}, q_{20}, q_{25}\}$

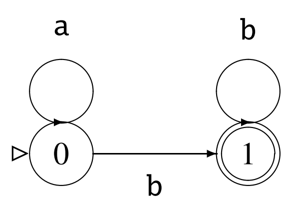
Und ferner ist $F := \{q_{25}\}$ sowie

(10.3) $\langle q_i, k, q_j\rangle \in \delta :\Leftrightarrow \begin{cases} i + k = j & \text{falls } i + k \leq 25, \\ j = 0 & \text{sonst.} \end{cases}$

Wir schreiben $j \xrightarrow{a} k$, falls $\langle j, a, k\rangle \in \delta$, und wir sagen, $\mathfrak{A}$ gehe mittels a von j in den Zustand k über. Es ist möglich, dass $j \xrightarrow{a} k'$ für ein $k \neq k'$. Es ist natürlich nicht ganz richtig, dass $\mathfrak{A}$ in einen einzelnen Zustand übergeht, denn es kann ja mehrere geben; aber die Formulierung ist praktischer.

Graphisch stellt man Automaten mit Hilfe von Kreisen und Pfeilen dar. Jeder Kreis symbolisiert einen Zustand, und zwischen den Zuständen hat man Pfeile, die mit Buchstaben gekennzeichnet sind. Falls von q nach q' ein Pfeil mit dem Buchstaben a geht, so bedeutet dies, dass $\langle q, a, q'\rangle \in \delta$. Die akzeptierenden Zustände werden durch einen Doppelkreis gekennzeichnet, der Initialzustand wird mit Hilfe einer kleinen Pfeilspitze oder $\rhd$ gekennzeichnet.

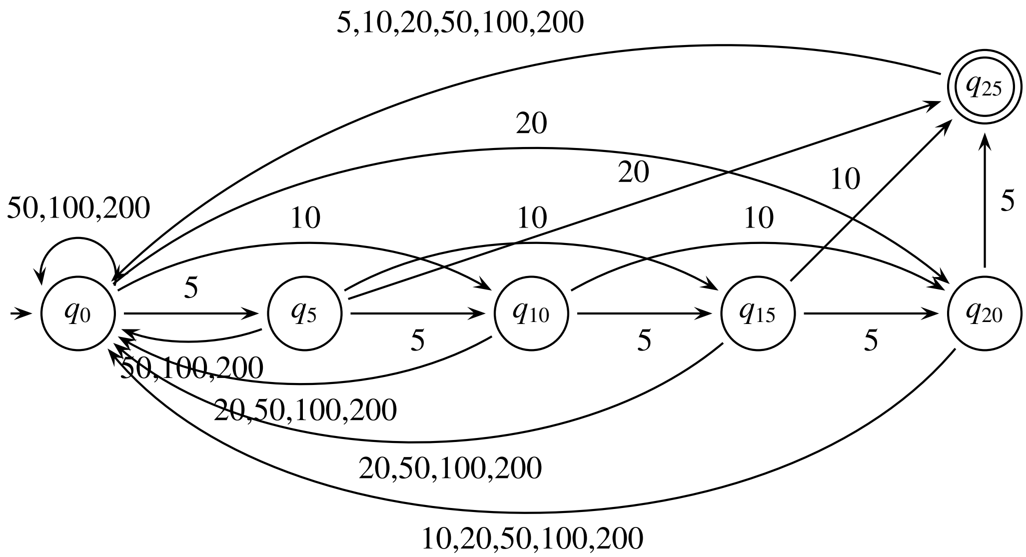
(10.4)



Dies ist ein Automat mit zwei Zuständen, 0 und 1. 0 ist der Initialzustand, 1 ist akzeptierend. Die Übergangsrelation ist $\{\langle 0, \mathsf{a}, 0\rangle, \langle 0, \mathsf{b}, 1\rangle, \langle 1, \mathsf{b}, 1\rangle\}$. Das Alphabet bleibt in dieser Zeichnung implizit und ist, wenn nichts anderes gesagt wird, die Menge aller in der Zeichnung vorkommenden Zeichen (also hier $\{\mathsf{a}, \mathsf{b}\}$). Hier ist also eine „formale" Beschreibung des Automaten:

(10.5) $\langle\{\mathsf{a}, \mathsf{b}\}, \{0, 1\}, 0, \{1\}, \{\langle 0, \mathsf{a}, 0\rangle, \langle 0, \mathsf{b}, 1\rangle, \langle 1, \mathsf{b}, 1\rangle\}\rangle$

Abbildung 10.1: Der Bezahlautomat



Kommen wir auf den eingangs erwähnten Automaten zurück. Dieser sieht wie in Figur 10.1 aus. Wir erweitern die Pfeilnotation wie folgt. Für ein Wort $\vec{x}$ schreiben wir

(10.6) $\quad i \xrightarrow{\varepsilon} j \;:\Leftrightarrow\; i = j$

(10.7) $\quad i \xrightarrow{a\cdot\vec{x}} j \;:\Leftrightarrow\;$ es existiert q mit $i \xrightarrow{a} q \xrightarrow{\vec{x}} j$

Für den eben definierten Automaten gilt, dass $0 \xrightarrow{\vec{x}} 1$ genau dann, wenn $\vec{x} = \mathtt{a}^m\mathtt{b}^n$ ist für $n > 0$. Diese Menge können wir durch $\mathtt{a}^*\mathtt{b}\mathtt{b}^*$ beschreiben. Sie ist also regulär.

Definition 10.2 (Lauf) *Ein **Lauf** des Automaten $\mathfrak{A}$ ist eine Folge*

(10.8) $\quad \ell = \langle q_0, a_0, q_1, a_1, q_2, \cdots, q_{n-1}, a_{n-1}, q_n\rangle,$

*wo q_0 der Startzustand ist, die a_i Buchstaben des Alphabets, und die q_i Zustände derart, dass $q_i \xrightarrow{a_i} q_{i+1}$. ℓ ist **akzeptierend**, falls $q_n \in F$. Wir sagen auch, dass ℓ ein Lauf über dem Wort $a_0a_1\cdots a_{n-1}$ ist. (Der Fall $n = 0$ ist zugelassen!)*

Den Lauf können wir auch wie folgt aufschreiben:

$$q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} q_2 \cdots \xrightarrow{a_{n-1}} q_n \tag{10.9}$$

Hier ist ein Lauf des obigen Automaten:

$$\langle q_0, 5, q_5, 10, q_{15}, 10, q_{25} \rangle \tag{10.10}$$

Dieser Lauf ist akzeptierend, denn er endet im Zustand q_{25} und $q_{25} \in F$. Alternativ notiert sieht der Lauf so aus:

$$q_0 \xrightarrow{5} q_5 \xrightarrow{10} q_{15} \xrightarrow{10} q_{25} \tag{10.11}$$

Hier nun ein Beispiel eines nicht akzeptierenden Laufs:

$$q_0 \xrightarrow{100} q_0 \xrightarrow{10} q_{10} \xrightarrow{5} q_{15} \tag{10.12}$$

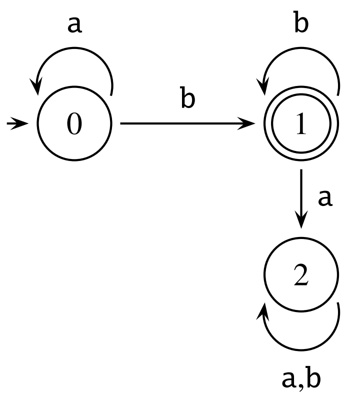
Nehmen wir nun den Automaten (10.4). Hier existieren Zeichenketten, für die es keinen Lauf gibt, so zum Beispiel /aaba/. Denn von 0 aus geht der Automat mit a nach 0, dann mit a nach 0, mit b nach 1. Jetzt ist noch ein a übrig, aber es gibt keinen Übergang von 1 aus mit a. Also existiert kein Lauf. Dies bedeutet übrigens, dass der Automat die Zeichenkette nicht akzeptiert.

Definition 10.3 (Sprache eines Automaten) *Ein endlicher Automat $\mathfrak{A}$ **akzeptiert** $\vec{x}$, falls es ein $q \in F$ gibt mit $i \xrightarrow{\vec{x}} q$. Die Menge aller von $\mathfrak{A}$ akzeptierten Wörter heißt die **Sprache** von $\mathfrak{A}$ und wird mit $L(\mathfrak{A})$ bezeichnet.*

Ein paar einfache Bemerkungen. Ist $F = \varnothing$, so ist $L(\mathfrak{A}) = \varnothing$. Es ist aber nicht unbedingt so, dass $L(\mathfrak{A}) = A^*$ ist, wenn nur $F = Q$. Denn dazu muss zu jedem Wort $\vec{x}$ erst einmal ein Zustand q existieren mit $i \xrightarrow{\vec{x}} q$. Ein endlicher Automat heißt **total**, wenn für jedes $q \in Q$ und $a \in A$ ein q' existiert mit $\langle q, a, q' \rangle \in \delta$. Offensichtlich existiert in einem totalen Automaten zu jedem $\vec{x}$ stets ein Zustand q mit $i \xrightarrow{\vec{x}} q$. Dies zeigt man zum Beispiel durch Induktion über die Länge von $\vec{x}$. Der Bezahlautomat zum Beispiel ist total, der Automat (10.4) dagegen nicht.

Wir können jeden endlichen Automaten total machen. Dazu müssen wir nur einen neuen Zustand μ hinzufügen (das heißt, ein Element $\mu \notin Q$). Dieser fängt alle Zeichenketten auf, für die es bisher keinen Lauf gab. Das geht, indem wir zu jedem Zustand q und Buchstaben a, zu dem noch kein Folgezustand existierte, nunmehr μ zum Folgezustand machen. Dabei darf man nicht vergessen, auch den neuen Zustand μ in die Rechnung einzubeziehen. Denn ist man in μ und kommt noch Buchstabe a herein, müssen wir einen Folgezustand haben. Dieser ist wiederum μ. Ich zeige dies zunächst an einem Beispiel, dem Automaten (10.4). Dieser ist partiell, weil es keinen Übergang von 1 mit dem Buchstaben a gibt. Wir fügen deswegen einen Zustand hinzu, den wir 2 nennen (der Name muss nicht μ sein), dazu den Übergang $\langle 1, \mathsf{a}, 2\rangle$ und die Übergänge $\langle 2, \mathsf{a}, 2\rangle$ und $\langle 2, \mathsf{b}, 2\rangle$.

(10.13)



Jetzt definieren wir ganz allgemein

(10.14) $\mathfrak{A}^+ := \langle A, Q \cup \{\mu\}, i, F, \delta^+\rangle$

Hierbei ist δ^+ wie folgt definiert.

(10.15)
$$\begin{aligned} \delta^+ := \quad & \delta \\ \cup \quad & \{\langle q, a, \mu\rangle : \text{ es existiert kein } q' \text{ mit } \langle q, a, q'\rangle \in \delta\} \\ \cup \quad & \{\langle \mu, a, \mu\rangle : a \in A\} \end{aligned}$$

Dies lässt sich kürzer auch so schreiben:

(10.16)
$$\begin{aligned} \delta^+ := \quad & \delta \\ \cup \quad & \{\langle q, a, \mu\rangle : (\{q\} \times \{a\} \times Q) \cap \delta = \varnothing\} \\ \cup \quad & \{\mu\} \times A \times \{\mu\} \end{aligned}$$

Für diesen Automaten ist $L(\mathfrak{A}^+) = L(\mathfrak{A})$.

Die Definition von $L(\mathfrak{A})$ besagt, dass eine Zeichenkette akzeptiert wird, sobald ein einziger akzeptierender Lauf existiert. Das bedeutet, dass eine Zeichenkette auch dann in $L(\mathfrak{A})$ ist, wenn es zusätzlich zu dem akzeptierenden Lauf auch noch nicht akzeptierende Läufe gibt. Um zu entscheiden, ob $\vec{x} \in L(\mathfrak{A})$ ist oder nicht, muss man also im Bedarfsfall *alle* Läufe für $\vec{x}$ ansehen. Es kann aber sehr viele davon geben. Ein Sonderfall aber besteht: wenn der Automat *deterministisch* ist, dann gibt es höchstens einen Lauf.

Definition 10.4 (Determinismus) *Ein Automat heißt **deterministisch**, falls gilt: ist* $q \xrightarrow{a} q'$ *und* $q \xrightarrow{a} q''$, *so gilt* $q' = q''$.

Ist ein Automat deterministisch, so existiert über einem gegebenen Wort höchstens ein Lauf. Sicherheitshalber formuliere ich dies in einem Satz.

Satz 10.5 *Es sei* $\mathfrak{A}$ *ein endlicher Automat,* q *ein Zustand und* $\vec{x}$ *eine beliebige Zeichenkette.*

1. *Ist* $\mathfrak{A}$ *total, so existiert ein* q' *mit* $q \xrightarrow{\vec{x}} q'$.
2. *Ist* $\mathfrak{A}$ *deterministisch, so existiert höchstens ein* q' *mit* $q \xrightarrow{\vec{x}} q'$.
3. *Ist* $\mathfrak{A}$ *deterministisch und total, so existiert genau ein* q' *mit* $q \xrightarrow{\vec{x}} q'$.

Der folgende Satz sichert uns auch noch, dass es zu jedem Automaten einen deterministischen und totalen Automaten gibt, der dasselbe leistet. Ich geben dazu ein Beispiel. Es sei der Automat $\langle Q, A, i_0, F, \delta\rangle$ gegeben, wo $A := \{\mathtt{a}, \mathtt{j}, \mathtt{x}\}$, $Q := \{0, 1, 2\}$, $i_0 := 0$, $F = \{0, 2\}$ und

$$\delta := \begin{array}{l}\{\langle 0, \mathtt{a}, 1\rangle, \langle 0, \mathtt{j}, 2\rangle, \langle 1, \mathtt{a}, 1\rangle, \langle 1, \mathtt{a}, 2\rangle, \\ \langle 1, \mathtt{x}, 0\rangle, \langle 2, \mathtt{j}, 2\rangle, \langle 2, \mathtt{x}, 1\rangle\}\end{array} \tag{10.17}$$

Betrachten wir die Zeichenkette /`jxa`/. Diese besitzt einen nicht akzeptierenden Lauf 0 `j` 2 `x` 1 `a` 1, und einen akzeptierenden Lauf 0 `j` 2 `x` 1 `a` 2. Folglich ist $\mathtt{jxa} \in L(\mathfrak{A})$.

Doch wie zählt man alle Läufe auf? Das Geheimnis ist, dass wir im letzten Schritt nicht einen einzigen Zustand hinschreiben sondern *alle* Folgezustände, also etwa so:

$$0\ \mathtt{j}\ 2\ \mathtt{x}\ 1\ \mathtt{a}\ \{1, 2\} \tag{10.18}$$

Betrachten wir nun ein Zeichenkette /`jxaxa`/. Um die Läufe hierfür aufzuzählen, müssen wir die Läufe für /`jxa`/ verlängern. Kommt der Buchstabe `x` und wir sind im Zustand 1, so ist der Folgezustand 0; sind wir in 2, so ist der Folgezustand 1. Wir listen diese auf:

$$0\ \mathtt{j}\ 2\ \mathtt{x}\ 1\ \mathtt{a}\ \{1, 2\}\ \mathtt{x}\ \{0, 1\} \tag{10.19}$$

Mittels a kommen wir mit a von 0 nach 1 und von 1 nach 1 oder 2. Der gesamte „Lauf" ist demnach

(10.20) $\quad 0\ \mathtt{j}\ 2\mathtt{x}\ 1\ \mathtt{a}\ \{1,2\}\ \mathtt{x}\ \{0,1\}\ \mathtt{a}\ \{1,2\}$

Der Clou ist, dass jeder in der Menge genannte Zustand über einen Lauf erreicht wird. Speziell also 2. Und weil $2 \in F$, wird die Zeichenkette akzeptiert.

Wir brauchen also statt mit einzelnen Zuständen nur mit Mengen von Zuständen zu arbeiten! Ist U solch eine Menge und $a \in A$, so ist der Folgezustand V die Menge aller Zustände q' für die ein $q \in U$ existiert mit $q \xrightarrow{a} q'$. Daraus können wir nun einen Automaten bauen. Im vorliegenden Fall hat der neue Automat 8 Zustände, nämlich $Q^p := \wp(\{0,1,2\})$. Die Übergangstabelle sieht so aus:

(10.21)

	a	j	x
$\varnothing$	$\varnothing$	$\varnothing$	$\varnothing$
$\{0\}$	$\{1\}$	$\{2\}$	$\varnothing$
$\{1\}$	$\{1,2\}$	$\varnothing$	$\{0\}$
$\{2\}$	$\varnothing$	$\{2\}$	$\{1\}$
$\{0,1\}$	$\{1,2\}$	$\{2\}$	$\{0\}$
$\{0,2\}$	$\{1\}$	$\{2\}$	$\{1\}$
$\{1,2\}$	$\{1,2\}$	$\{2\}$	$\{0,1\}$
$\{0,1,2\}$	$\{1,2\}$	$\{2\}$	$\{0,1\}$

Der Startzustand ist $\{0\}$; und schließlich sind die finalen Zustände alle solchen, die 0 oder 2 enthalten, also setzen wir $F^p := \{\{0\},\{2\},\{0,1\},\{1,2\},\{0,1,2\}\}$. Dies definiert nun einen neuen Automaten mit 8 Zuständen, der deterministisch ist und dieselbe Sprache hat wie $\mathfrak{A}$. Der Lauf zu der Zeichenkette /jxaxa/ ist nun wie folgt:

(10.22) $\quad \{0\}\ \mathtt{j}\ \{2\}\mathtt{x}\ \{1\}\ \mathtt{a}\ \{1,2\}\ \mathtt{x}\ \{0,1\}\ \mathtt{a}\ \{1,2\}$

Weil $\{1,2\} \in F^p$, wird die Zeichenkette akzeptiert.

Satz 10.6 *Zu jedem endlichen Automaten $\mathfrak{A}$ existiert ein deterministischer, totaler Automat $\mathfrak{A}^p$ mit $L(\mathfrak{A}^p) = L(\mathfrak{A})$.*

Beweis. Es sei $\mathfrak{A} = \langle A, Q, i, F, \delta\rangle$. Dann setze $Q^p := \wp(Q)$, $i^p := \{i\}$, und $F^p := \{S \subseteq Q : S \cap F \neq \varnothing\}$. Und schließlich sei

(10.23) $\quad S \xrightarrow{a} U$ gdw. $U = \{q' : \text{es existiert ein } q \in S \text{ mit } q \xrightarrow{a} q'\}$

Mit anderen Worten: U ist die Menge aller Zustände, die man von einem Zustand aus S mittels a erreichen kann. Dies kann man auch so schreiben:

(10.24) $\qquad U := (S \times \{a\} \times Q) \cap \delta$

Es sei also

(10.25) $\qquad \delta^p := \{\langle U, a, V\rangle : U \stackrel{a}{\to} V\}$

Wir setzen nun

(10.26) $\qquad \mathfrak{A}^p := \langle A, Q^p, i^p, F^p, \delta^p\rangle$

Da U eindeutig durch S und a bestimmt ist, ist der Automat deterministisch. Er ist auch total, denn U existiert immer. (U darf auch leer sein!) Wir zeigen jetzt folgende Behauptung induktiv über die Länge von $\vec{x}$:

(10.27) $\qquad S \xrightarrow{\vec{x}} U$ gdw. $U = \{q' :$ es existiert ein $q \in S$ mit $q \xrightarrow{\vec{x}} q'\}$

Daraus folgt dann schon die Behauptung des Satzes, was man wie folgt sieht. Sei $\vec{x} \in L(\mathfrak{A})$. Dann existiert ein $q \in F$ mit $i \xrightarrow{\vec{x}} q$. Es sei U nun diejenige Menge, für die gilt $i^p = \{i\} \xrightarrow{\vec{x}} U$. Dann ist wegen (10.27) $q \in U$, und da $q \in F$, so ist $U \in F^p$. Zusammen haben wir, dass $\vec{x} \in L(\mathfrak{A}^p)$. Ist umgekehrt $\vec{x} \in L(\mathfrak{A}^p)$, so existiert ein $U \in F^p$ mit $\{i\} \xrightarrow{\vec{x}} U$. Nach Definition ist für jedes $q \in U$ $i \xrightarrow{\vec{x}} q$. Es existiert ein $q' \in U \cap F$, und so ist insbesondere $i \xrightarrow{\vec{x}} q'$, also $\vec{x} \in L(\mathfrak{A})$.

Nun zum Beweis von (10.27). Es sei n die Länge von $\vec{x}$. Für $n = 0$ ist Behauptung klar. Denn nach Definition (10.23) ist $S \xrightarrow{\varepsilon} U$ gdw. $U = S$, und dann ist U tatsächlich die Menge der von einem Zustand in S aus erreichbaren Zustände.

Sei nun die Behauptung für alle Zahlen $< n + 1$ gezeigt und habe $\vec{x}$ die Länge $n+1$. Dann ist $\vec{x} = \vec{y} \cdot a$ für ein a und ein $\vec{y}$. Es sei V so gewählt, dass $S \xrightarrow{\vec{y}} V \xrightarrow{a} U$. Nach Voraussetzung ist V die Menge der von einem Zustand in S aus mit $\vec{y}$ erreichbaren Zustände. Ist nun $p \xrightarrow{\vec{x}} q'$ für ein q', so ist $p \xrightarrow{\vec{y}} q'' \xrightarrow{a} q'$ für ein q''. Nach Induktionsannahme ist $q'' \in V$. Deswegen ist jeder Zustand in U mittels a von einem Zustand in V aus erreichbar. Und so ist U die Menge der von S mit $\vec{x}$ erreichbaren Zustände. ⊣

Wir können daraus einen wichtigen Schluss ziehen.

Korollar 10.7 *Ist S eine durch einen endlichen Automaten akzeptierte Sprache, so auch $A^* - S$.*

Beweis. Wir können wegen dem Vorangegangenen annehmen, dass es einen deterministischen und totalen Automaten $\mathfrak{A} = \langle A, Q, i, F, \delta\rangle$ gibt mit $S = L(\mathfrak{A})$. Sei $\mathfrak{A}^n := \langle A, Q, i, Q - F, \delta\rangle$. Ich behaupte, dass $L(\mathfrak{A}^n) = A^* - S$. Dazu sei $\vec{x} \in A^*$. Da die Übergangsfunktion in beiden Automaten gleich ist und total und deterministisch, existiert nach Satz 10.5 ein eindeutiges q mit $i \xrightarrow{\vec{x}} q$. Es ist $\vec{x} \in L(\mathfrak{A})$ genau dann, wenn $q \in F$ und $\vec{x} \in L(\mathfrak{A}^n)$ genau dann, wenn $q \in Q - F$, dh genau dann, wenn $q \notin F$. Dies zeigt die Behauptung. ⊣

Satz 10.8 *Genau dann ist eine Sprache S regulär, wenn es einen endlichen Automaten $\mathfrak{A}$ gibt mit $L(\mathfrak{A}) = S$.*

Der Beweis ist recht langwierig und sei hier nicht erbracht. Anstelle dessen gebe ich Beispiele. So etwa ein Automat, der nur ein einziges Wort erkennt, etwa das Wort /`Katze`/. Seine Zustände sind q_ε, $q_{\texttt{K}}$, $q_{\texttt{Ka}}$, $q_{\texttt{Kat}}$, $q_{\texttt{Katz}}$ und $q_{\texttt{Katze}}$. Und es gibt lediglich die folgenden Übergänge, die ich der Übersicht halber verkürzt darstelle:

$$(10.28) \qquad q_\varnothing \xrightarrow{\texttt{K}} q_{\texttt{K}} \xrightarrow{\texttt{a}} q_{\texttt{Ka}} \xrightarrow{\texttt{t}} q_{\texttt{Kat}} \xrightarrow{\texttt{z}} q_{\texttt{Katz}} \xrightarrow{\texttt{e}} q_{\texttt{Katze}}$$

Und schließlich ist $F := \{q_{\texttt{Katze}}\}$. Es sollte klar sein, wie man das Rezept auf ein beliebiges Wort ausdehnt.

Übungen

[47] Es sei $A = \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}$. Es sei ein Automat $\mathfrak{A}$ mit den Zuständen 0, 1 und 2 gegeben, wobei 0 initial, 1 und 2 akzeptierend sind. Die Menge der Übergänge sei

$$\{\langle 0, \mathsf{a}, 1\rangle, \langle 1, \mathsf{b}, 1\rangle, \langle 1, \mathsf{c}, 2\rangle, \langle 2, \mathsf{a}, 1\rangle, \langle 2, \mathsf{a}, 0\rangle\}$$

Zeichnen Sie $\mathfrak{A}$! Ist dieser Automat (a) deterministisch, (b) total?

[48] Finden Sie eine Zeichenkette, für die $\mathfrak{A}$ einen akzeptierenden Lauf und einen nicht akzeptierenden Lauf besitzt und geben Sie beide an. Ist diese Zeichenkette in der Sprache des Automaten?

Kapitel 11

Reguläre Grammatiken

In diesem Kapitel geben wir noch eine dritte Charakterisierung der regulären Sprachen: die regulären Sprachen sind diejenigen, die eine rechtsreguläre Grammatik haben.

Definition 11.1 (Reguläre Grammatik) *Eine Grammatik G heißt **rechtsregulär** oder einfach nur **regulär**, wenn die Regeln die Form $X \to a$, $X \to \varepsilon$ oder $X \to aY$ haben, wo X und Y beliebige Nichtterminalsymbole sind und a ein Terminalsymbol.*

Hier ist zum Beispiel eine Grammatik für die Sprache $(\mathtt{ab})^+$: $A := \{\mathtt{a}, \mathtt{b}\}$, $N := \{\mathtt{S}, \mathtt{T}\}$, S das Startsymbol, und die Regeln sind

(11.1) $\mathtt{S} \to \mathtt{aT}, \quad \mathtt{T} \to \mathtt{b} \mid \mathtt{bS}$

(Zur Notation '|' siehe Seite (8.25).) Hier ist eine Ableitung von /ababab/:

(11.2) $\mathtt{S}, \mathtt{aT}, \mathtt{abS}, \mathtt{abaT}, \mathtt{ababS}, \mathtt{ababaT}, \mathtt{ababab}$

Hier noch eine weitere Grammatik, die nur eine einzige Zeichenkette erzeugen kann, nämlich /Hund/. Es sei $A := \{\mathtt{a}, \cdots, \mathtt{z}, \mathtt{H}\}$ (Achtung: H ist terminal!) und $N := \{\mathtt{S}, \mathtt{T}, \mathtt{U}, \mathtt{V}\}$. Es sei S das Startsymbol.

(11.3)
$$\begin{array}{ll} \mathtt{S} \to \mathtt{HT} & \mathtt{T} \to \mathtt{uU} \\ \mathtt{U} \to \mathtt{nV} & \mathtt{V} \to \mathtt{d} \end{array}$$

Hier also eine Ableitung von /Hund/, welche sogar die einzige ist:

(11.4) $\mathtt{S}, \mathtt{HT}, \mathtt{HuU}, \mathtt{HunV}, \mathtt{Hund}$

Wer die Grammatik etwas besser aufschreiben möchte, kann sich der spitzen Klammern bedienen. Ich mache dies exemplarisch vor. A sei jetzt unser normales Alphabet mit Klein- und Großbuchstaben, $N := \{\texttt{<0>}, \texttt{<1>}, \texttt{<2>}, \texttt{<3>}\}$. Das Startsymbol ist $\texttt{<0>}$.

(11.5)
$$\begin{array}{ll} \texttt{<0>} \to \texttt{H<1>} & \texttt{<1>} \to \texttt{u<2>} \\ \texttt{<2>} \to \texttt{n<3>} & \texttt{<3>} \to \texttt{d} \end{array}$$

Die Ableitung von /Hund/ (analog zu (11.4)) sieht jetzt so aus:

(11.6) $\texttt{<0>}, \texttt{H<1>}, \texttt{Hu<2>}, \texttt{Hun<3>}, \texttt{Hund}$

Mittels der Zahlen zählen wir einfach, wie viele Symbole bereits ausgegeben wurden. Damit ist wiederum klar, wie wir eine reguläre (!) Grammatik schreiben, die ein festes Wort ausgibt und sonst nichts.

Es ist kein Zufall, dass es immer nur ein einziges Nichtterminalsymbol in den Zeichenketten gibt und dieses am rechten Rand auftritt.

Proposition 11.2 *Es sei G eine reguläre Grammatik und $\vec{x} \in (A \cup N)^*$ ein Wort mit $\vdash_G \vec{x}$. Dann ist $\vec{x} \in A^* \cup A^* \cdot N$. Das heißt, $\vec{x}$ hat die Form $a_0 a_1 \cdots a_{n-1}$ oder $a_0 a_1 \cdots a_{n-1} B$, wo die a_i terminale Buchstaben sind und B ein Nichtterminalzeichen.*

Dies kann man sich leicht selbst überlegen; dazu bedienen wir uns der Induktion über die Länge der Ableitung. Ist diese 0, so haben wir nur das Anfangswort, und dieses besteht aus dem Startsymbol. Dies erfüllt offensichtlich die Behauptung. Nun habe die Ableitung die Länge $n + 1$ und die Behauptung sei schon für alle Ableitungen der Länge $\leq n$ gezeigt. Dann ist das Ende der Ableitung wie folgt: $\langle \cdots, \vec{x}B, ? \rangle$. Die letzte Zeichenkette wird durch Ersetzung von B erzeugt. Dabei gibt es zwei Fälle. (a) Wir nehmen eine Regel der Form $B \to a$ oder $B \to \varepsilon$. Dann erhalten wir das Wort $\vec{x}a$ bzw. $\vec{x}$. (b) Wir nehmen eine Regel der Form $B \to aC$, dann erhalten wir ein Wort der Form $\vec{x}aC$. In beiden Fällen gilt die Behauptung.

Ich weise darauf hin, dass in dem Beweis wesentlich verwendet wurde, dass $\vec{x}$ aus S abgeleitet wurde. Nehmen wir zum Beispiel die obige Grammatik. Dann gilt `TabS` $\vdash_G$ `TabaT`, und `TabaT` hat nicht die versprochene Form (und kann deswegen nicht aus dem Startsymbol hergeleitet werden). Allerdings hat auch schon `SabT` nicht diese Form, und wenn man den Beweis genau anschaut, sieht man, dass immer wenn $\vec{x}$ die gewünschte Form hat und $\vec{x} \vdash_G \vec{y}$, dann ist auch $\vec{y}$ von dieser Form.

Die Äquivalenz zwischen endlichen Automaten und regulären Grammatiken ist relativ direkt zu sehen. Dazu erst einmal eine wichtige Beobachtung.

Proposition 11.3 *Zu jeder regulären Grammatik G existiert eine reguläre Grammatik G° mit $L(G^\circ) = L(G)$, welche keine Regeln der Form $A \to a$ besitzt.*

Beweis. Sei $G = \langle A, N, S, R\rangle$. Sei $E \notin N$; wir ersetzen jede Regel der Form $A \to a$ durch zwei Regeln, nämlich $A \to aE$ und $E \to \varepsilon$. Genauer ist

$$(11.7)\qquad \begin{aligned} R^\circ := \quad & \{A \to aE : A \to a \in R\} \\ \cup\ & \{A \to aB : A \to aB \in R\} \\ \cup\ & \{E \to \varepsilon\} \end{aligned}$$

Schließlich ist

$$(11.8)\qquad G^\circ := \langle A, N \cup \{E\}, S, R^\circ\rangle$$

Zu zeigen ist $L(G^\circ) = L(G)$. Betrachten wir dazu eine G°-Ableitung $\langle S, \vec{x}_1, \vec{x}_2, \cdots, \vec{x}_n\rangle$, wo $\vec{x}_n \in A^*$. Setze $\vec{x}_0 := S$. Im Übergang von $\vec{x}_i$ nach $\vec{x}_{i+1}$ wurde für $i < n - 1$ eine Regel der Form $A \to aB$ verwendet. Diese ist auch in G. Wir müssen also nur den Fall $i = n - 2$ ansehen. Die letzte Regel hat dann die Form $U \to \varepsilon$. Dann ist $\vec{x}_{n-1} = \vec{x}_n U$ für ein $U \in N \cup \{E\}$. Ist $U \notin E$, so ist $U \to \varepsilon \in R$, und die Ableitung ist eine G-Ableitung. Sei nun $U = E$. Dann ist $n - 1 > 0$, weil ja $E \neq S$ ist. Die vorletzte Regelanwendung hatte also die Form $V \to aE$. Die Ableitung endet also wie folgt $\langle ..., \vec{y}V, \vec{y}aE, \vec{y}a\rangle$. Nach Konstruktion von R° ist dann $V \to a \in R$, also ist $\langle \vec{x}_0, \cdots, \vec{x}_{n-3}, \vec{y}V, \vec{y}a\rangle$ eine G-Ableitung. Also ist $L(G^\circ) \subseteq L(G)$. Nun sei $\langle \vec{x}_0, \cdots, \vec{x}_n\rangle$ eine G-Ableitung und $\vec{x}_n \in A^*$. Dann ist $\vec{x}_{n-1} = \vec{y}U$ und $\vec{x}_n = \vec{y}a$ für ein $U \in N$ und $a \in A$. Wiederum sind alle Schritte bis auf den letzten auch G°-Schritte. Wir ändern diese Ableitung nun wie folgt: $\langle \vec{x}_0, \cdots, \vec{y}U, \vec{y}aE, \vec{y}a\rangle$. Dies ist eine G°-Ableitung. ⊣

Für die Grammatik (11.1) liefert die Konstruktion die folgende Grammatik:

$$(11.9)\qquad \mathtt{S} \to \mathtt{aT}, \quad \mathtt{T} \to \mathtt{bE} \mid \mathtt{bS}, \quad \mathtt{E} \to \varepsilon$$

Wir konstruieren zu einer solchen Grammatik einen endlichen Automaten auf die folgende Weise. Die Zustände sind jetzt N, der Startzustand ist S, und wir sagen, dass genau dann $A \xrightarrow{a} B$, wenn die Grammatik die Regel $A \to aB$ besitzt. Mit anderen Worten,

$$(11.10)\qquad \delta := \{\langle A, a, B\rangle : A \to aB \in R\}$$

Ferner ist

$$(11.11)\qquad F := \{A : A \to \varepsilon \in R\}$$

Dies definiert den Automaten.

Und damit erhalten wir aus (11.9) den folgenden Automaten

(11.12) $\langle\{\mathtt{a},\mathtt{b}\},\{\mathtt{S},\mathtt{T},\mathtt{E}\},\mathtt{S},\{\mathtt{E}\},\delta\rangle$

Hier ist

(11.13) $\delta = \{\langle\mathtt{S},\mathtt{a},\mathtt{T}\rangle,\langle\mathtt{T},\mathtt{b},\mathtt{E}\rangle,\langle\mathtt{T},\mathtt{b},\mathtt{S}\rangle\}$

Man beachte, dass der Automat nicht deterministisch ist. (Man kann aber einen deterministischen Automaten konstruieren, der die gleiche Sprache erkennt, wie wir gesehen haben.)

Es gibt eine eins-zu-eins Abbildung zwischen Ableitungen und Läufen des Automaten. Die Ableitung beginnt mit dem Startsymbol S, der Lauf mit dem Startzustand, und dieser ist auch S. Im nächsten Schritt wählen wir eine Regel aus (etwa $\mathtt{S} \to \mathtt{aT}$), mittels derer wir die neue Zeichenkette erhalten, hier /aT/. Für den Automaten heißt dies, der Lauf wird um das Symbol /a/ und den neuen Zustand T verlängert. Anschließend ersetzen wir /T/ durch /bS/, und analog verlängern wir den Lauf um /b/ gefolgt von /S/. Und so machen wir weiter. Am Ende, wenn wir /T/ durch /b/ ersetzen, verlängern wir den Lauf um /b/ gefolgt von /E/. Der Lauf, der zu (11.2) korrespondiert, ist hier wiedergegeben:

(11.14) $\langle\mathtt{S},\mathtt{a},\mathtt{T},\mathtt{b},\mathtt{S},\mathtt{a},\mathtt{T},\mathtt{b},\mathtt{S},\mathtt{a},\mathtt{T},\mathtt{b},\mathtt{E}\rangle$

Nun sei ein Lauf des Automaten gegeben. Dieser hat die Form $\langle i, a_0, q_0, a_1, q_1, \cdots\rangle$. Nach Konstruktion des Automaten ist dann $i = S$, das Startsymbol, und die q_i gewisse Nichtterminale B_i. Wir formen daraus diese Ableitung:

(11.15) $\langle S, a_0B_0, a_0a_1B_1, a_0a_1a_2B_2, \cdots\rangle$

Ist der Lauf akzeptierend, so endet er in E, und dann erhalten wir diese Ableitung:

(11.16) $\langle S, a_0B_0, a_0a_1B_1, a_0a_1a_2B_2, \cdots, a_0a_1\cdots a_{n-1}\rangle$

Wenn wir dieses Rezept auf (11.14) anwenden, erhalten wir (11.2) zurück.

Nun sei $\mathfrak{A} = \langle A, Q, i, F, \delta\rangle$ ein endlicher Automat. Dann setzen wir $N := Q$ (die Nichtterminalsymbole der Grammatik sind die Zustände des Automaten), das Startsymbol ist i, und wir haben folgende Regeln:

(11.17) $R := \{q \to \varepsilon : q \in F\} \cup \{q \to aq' : (q, a, q') \in \delta\}$

Ein Lauf des Automaten lässt sich wie oben eins-zu-eins in eine Ableitung der zugehörigen Grammatik übersetzen. Als Beispiel nehme ich den Automaten (10.4). Dieser hat zwei Zustände, 0 und 1, denen ich die Nichtterminalzeichen $\texttt{<0>}$ und $\texttt{<1>}$ zuordne. Das Startymbol is $\texttt{<0>}$, weil 0 initialer Zustand ist. Die Regeln sind

$$\begin{aligned} &\texttt{<0>} \to \texttt{a<0>} \\ &\texttt{<0>} \to \texttt{b<1>} \\ &\texttt{<1>} \to \texttt{b<1>} \\ &\texttt{<1>} \to \varepsilon \end{aligned} \tag{11.18}$$

Die ersten drei Regeln bilden die Übergänge ab. Die letzte Regel ist aufgenommen worden, weil $\texttt{<1>}$ akzeptierend ist. Diese Grammatik kann wie folgt aufgeschrieben werden.

$$\begin{aligned} &\texttt{<0>} \to \texttt{a<0>} \mid \texttt{b<1>} \\ &\texttt{<1>} \to \texttt{b<1>} \mid \varepsilon \end{aligned} \tag{11.19}$$

Wir bezeichnen den zu der Grammatik G konstruierten Automaten mit $A(G)$, und den zu dem Automaten $\mathfrak{A}$ konstruierte Grammatik mit $\Gamma(\mathfrak{A})$.

Satz 11.4 *Für eine beliebe reguläre Grammatik ist $L(\Gamma(A(G))) = L(G)$; für einen endlichen Automaten $\mathfrak{A}$ ist $L(A(\Gamma(\mathfrak{A}))) = L(\mathfrak{A})$. Daraus folgt unmittelbar, dass die Klasse der durch reguläre Grammatiken erzeugten Sprachen genau die Klasse der regulären Sprachen ist.*

Man beachte, dass in dem Satz nicht gesagt wird, dass $\Gamma(A(G)) = G$. Dies ist nämlich gar nicht der Fall. Sei nämlich $G = \langle A, N, S, R\rangle$ gegeben. Dann ist $A(G) = \langle A, N \cup \{\texttt{E}\}, S, \{\texttt{E}\}, \delta\rangle$. Daraus wird dann $\Gamma(A(G)) = \langle A, N \cup \{\texttt{E}\}, S, R'\rangle$ für ein gewisses R'. Man beachte insbesondere, dass die Konstruktion des Automaten einen neuen Zustand erforderte, der nun in der Rückübersetzung als neues Nichtterminalsymbol firmiert. Aber wenn man genauer hinsieht, entdeckt man, dass die Unterschiede vernachlässigbar sind. Dazu nehmen wir uns mal die Regeln bzw. Übergänge vor. Es seien Regeln R gegeben. Der konstruierte Automat hat die Übergänge $A \xrightarrow{a} \texttt{E}$, wenn $A \to a$ eine Regel von G ist, und er hat die Übergänge $A \xrightarrow{a} B$, wenn $A \to aB$ eine Regel ist. Nun schauen wir uns die Regeln von $\Gamma(A(G))$ an. $\texttt{E}$ ist der einzige akzeptierende Zustand. Ist dann $A \xrightarrow{a} \texttt{E}$ ein Übergang des Automaten, so wird $\Gamma(A(G))$ die Regel $A \to a$ enthalten. Ist $A \xrightarrow{a} B$ ein Übergang und B nicht akzeptierend, also $B \neq \texttt{E}$, dann ist $A \xrightarrow{a} B$ eine Regel von $\Gamma(A(G))$. Also haben G und $\Gamma(A(G))$ dieselben Regeln.

Betrachten wir nun einen Automaten $\mathfrak{A}$. Wiederum gilt nicht unbedingt $\mathfrak{A} = A(\Gamma(\mathfrak{A}))$. Denn man beachte: während $\mathfrak{A}$ beliebig viele akzeptierende Zustände haben kann, hat $A(\Gamma(\mathfrak{A}))$ stets nur einen. Also können wir nicht erwarten, denselben Automaten zurückzubekommen. In dem Fall, wo $\varepsilon \notin L(\mathfrak{A})$, kann man aber jeden Automaten umtransformieren in einen Automaten, der dieselbe Sprache erkennt und nur einen akzeptierenden Zustand. Die Idee dabei ist, dass man einen neuen Zustand $\mathtt{E}$ dazugibt und für jeden Übergang $q \xrightarrow{a} q'$ für $q' \in F$ noch einen Übergang $q \xrightarrow{a} \mathtt{E}$. Anschließend ersetzen wir die ursprüngliche Menge F durch $\{\mathtt{E}\}$. Dieser Automat wiederum hat die Eigenschaft, dass $A(\Gamma(\mathfrak{A})) = \mathfrak{A}$, und daraus ergibt sich letztlich die Behauptung des obigen Satzes.

Übungen

[49] Gegeben sei der Automat $\mathfrak{A}$ aus Übung 47. Konstruieren Sie eine reguläre Grammatik G, die die Sprache des Automaten $\mathfrak{A}$ erzeugt.

[50] Leiten Sie in der Grammatik G die Zeichenketten `abbc` und `acaa` ab.

Kapitel 12

Kontextfreie Grammatiken und Satzstruktur

In diesem Kapitel werden wir uns mit Strukturen zur Beschreibung von Sätzen befassen. Hierbei geht es Strukturbeschreibung im Zusammenhang mit kontextfreien Grammatiken. Wir werden sehen, dass eine kontextfreie Grammatik einem ableitbaren Satz auch eine Struktur zuschreibt. Dies ist die Struktur eines geordneten Baumes, welche vom sprachlichen Standpunkt aus viel wichtiger ist als die Ableitung, weil sie etwas über die Bedeutung aussagt.

Es sei A ein Alphabet terminaler Zeichen, N ein Alphabet nichtterminaler Zeichen. Variable für terminale Zeichen sind Kleinbuchstaben (a, b und so weiter), Variable für Nichtterminalzeichen sind Großbuchstaben (also X, Y und so weiter), und kleine griechische Buchstaben bezeichnen ein beliebiges Element aus $A \cup N$. Pfeile verwende ich wie üblich, um Zeichenketten zu kennzeichnen.

Definition 12.1 (Kontextfreie Grammatik und Sprache) *Eine **kontextfreie Regel** ist eine Regel der Form $X \to \vec{\gamma}$, wo X ein Nichtterminalsymbol ist und $\vec{\gamma}$ eine beliebige Zeichenkette. Eine Grammatik G heißt **kontextfrei** oder kurz eine **KFG**, wenn jede Regel kontextfrei ist. Eine Sprache S heißt **kontextfrei**, wenn es eine kontextfreie Grammatik G gibt mit $L(G) = S$.*

Im Allgemeinen wird noch ein klein wenig mehr verlangt, damit eine Grammatik kontextfrei genannt wird, nämlich dass $\vec{\gamma}$ nicht das leere Wort ist, es sei denn, A ist das Startsymbol (also S). Und in diesem Fall darf keine Regel existieren der Form $B \to \vec{\eta}$, wo $\vec{\eta}$ S enthält. Diese Feinheit soll uns aber nicht interessieren, denn sie wird nur benötigt, wenn die Sprache auch ε enthält, ein Fall, der eigentlich nur von theoretischem Interesse ist. Ich weise aber darauf hin, dass, auch wenn wir diese Bedingung weglassen, die Grammatiken technisch gesehen nicht kontextfrei sind; aber erstens erzeugen sie kontextfreie Sprachen, zweitens ist die Abweichung minimal.

Kontextfreie Sprachen können auch durch scheinbar sehr komplexe (insbesondere nicht kontextfreie) Grammatiken erzeugt werden. Ich gebe ein Beispiel.

(12.1)
$$\begin{array}{ll} \mathtt{S} & \to \mathtt{xAy} \\ \mathtt{A} & \to \mathtt{xAy/x}\_\_ \\ \mathtt{A} & \to \mathtt{c} \end{array}$$

Die zweite Regel ist nicht kontextfrei. Sie beschreibt eine kontextgebundene Ersetzung. Wir können dies sehen, wenn wir die Kontextbedingung eliminieren; dann bekommen wir nämlich die Regel $\mathtt{xA} \to \mathtt{xxAy}$. Und diese ist klarerweise nicht kontextfrei. Allerdings kann man leicht sehen, dass das Symbol `A` immer in dem geforderten Kontext vorkommt, sodass die Grammatik (12.1) die gleiche Sprache erzeugt wie (12.2).

(12.2)
$$\begin{array}{ll} \mathtt{S} & \to \mathtt{xAy} \\ \mathtt{A} & \to \mathtt{xAy} \\ \mathtt{A} & \to \mathtt{c} \end{array}$$

Diese Grammatik ist kontextfrei und erzeugt die Sprache $\{\mathtt{x}^n\mathtt{cy}^n : n \in \mathbb{N}, n > 0\}$.

Zunächst einmal sei Folgendes hervorgehoben.

Proposition 12.2 *Reguläre Grammatiken sind kontextfrei.*

Dies ist so, weil eine reguläre Regel auch eine kontextfreie Regel ist.

Kontextfreie Grammatiken sind eine sehr wichtige Klasse von Grammatiken. Computersprachen sind in der Regel in KFGs verfasst. (Dies geschieht überlicherweise in der Backus-Naur-Form, die sich von der hier gezeigten Form nur dadurch unterscheidet, dass sie anstelle von → die Kette ::= verwendet.) Eine kontextfreie Grammatik erlaubt auch, den Zeichenketten eine Struktur zuzuweisen. Bevor ich dies erklären kann, muss ich auf den Begriff der Ableitung zurückkommen.

Nehmen wir noch einmal die Grammatik von Kapitel 8.

(12.3)
$$\begin{array}{ll} \mathtt{S}\to\mathtt{UP} & \mathtt{U}\to\mathtt{DN} \mid \mathtt{DNR} \\ \mathtt{D}\to\mathtt{the}\text{␣} \mid \mathtt{a}\text{␣} & \mathtt{N}\to\mathtt{car}\text{␣} \mid \mathtt{rat}\text{␣} \mid \mathtt{mouse}\text{␣} \\ \mathtt{R}\to\mathtt{TP} & \mathtt{T}\to\mathtt{that}\text{␣} \\ \mathtt{P}\to\mathtt{I} \mid \mathtt{VU} & \mathtt{I}\to\mathtt{ran}\text{␣} \mid \mathtt{drove}\text{␣} \\ \mathtt{V}\to\mathtt{drove}\text{␣} \mid \mathtt{chased}\text{␣} & \end{array}$$

Diese ist kontextfrei, wie man leicht sieht. Sie hat zum Beispiel die Regel $\mathtt{S} \to \mathtt{UP}$. Diese sagt, dass die nach ihrer Verwendung erzeugte Zeichenkette $\vec{x}$ in zwei Teile zerfällt, nämlich $\vec{x} = \vec{y}\vec{z}$, wo $\mathtt{U} \vdash_G \vec{y}$ und $\mathtt{P} \vdash_G \vec{z}$.

Ich werde diese Grammatik etwas freundlicher gestalten. Es sei jetzt A unser normales Alphabet der Kleinbuchstaben und dem Leerzeichen. Ferner sei

(12.4) N = { <Satz>, <Det>, <DetP>, <RelS>,
<Präd>, <Nomen>, <TVerb>, <Comp>, <IVerb>}

Das Startsymbol sei <Satz>, und R bestehe aus den folgenden Regeln:

(12.5)

<Satz>→<DetP><Präd>
<DetP>→<Det><Nomen> | <Det><Nomen><RelS>
<Det>→the␣ | a␣
<Nomen>→car␣ | rat␣ | mouse␣
<RelS>→<Comp><Präd>
<Comp>→that␣
<Präd>→<IVerb> | <TVerb><DetP>
<IVerb>→ran␣ | drove␣
<TVerb>→drove␣ | chased␣

Mit dieser Grammatik leiten wir nun erneut denselben Satz ab:

(12.6)

1. <Satz>
2. <DetP><Präd>
3. <Det><Nomen><RelS><Präd>
4. the␣<Nomen><RelS><Präd>
5. the␣rat␣<RelS><Präd>
6. the␣rat␣<Comp><Präd><Präd>
7. the␣rat␣that␣<Präd><Präd>
8. the␣rat␣that␣<TVerb><DetP><Präd>
9. the␣rat␣that␣drove␣<DetP><Präd>
10. the␣rat␣that␣drove␣<Det><Nomen><Präd>
11. the␣rat␣that␣drove␣a␣<Nomen><Präd>
12. the␣rat␣that␣drove␣a␣car␣<Präd>
13. the␣rat␣that␣drove␣a␣car␣<IVerb>
14. the␣rat␣that␣drove␣a␣car␣ran␣

Wir sagen, dass diese Ableitung dem Satz eine bestimmte Struktur gibt. Ich gehe dies schrittweise durch. Als erstes sagt die Ableitung, dass unsere Zeichenkette ein Satz ist. Dies notieren wir so.

(12.7) [<Satz>the␣rat␣that␣drove␣a␣car␣ran␣]

Im ersten Schritt haben wir im Übergang von Zeile 1 zu Zeile 2 das Symbol <Satz> durch die Kette von zwei Symbolen <DetP><Präd> ersetzt. Die Regel deuten wir nun wie folgt. Eine beliebige Zeichenkette $\vec{x}$, welche aus <Satz> in dieser Grammatik abgeleitet werden kann, hat eine Zerlegung $\vec{x} = \vec{y} \cdot \vec{z}$, wo $\vec{y}$ aus <DetP> und $\vec{z}$ aus <Präd> abgeleitet ist. Das soll heißen: Wie auch immer die Ableitung weitergeht, die Zeichenkette zerfällt so in zwei Teile, deren erster eine Determinatorphrase (<DetP>) und deren zweiter ein Prädikat (<Präd> ist. Wir stellen das bildlich auf zwei Weisen dar. Die erste ist, dass wir die Zeichenkette mit Klammern versehen, die die Teilung offenbar werden lassen.

(12.8) [<Satz>[<DetP>the␣rat␣that␣drove␣a␣car␣][<Präd>ran␣]]

Oder auch, ohne Nichtterminalsymbole:

(12.9) [[the␣rat␣that␣drove␣a␣car␣][ran␣]]

Im zweiten Schritt wird <DetP> durch die Folge <Det><Nomen><RelS> ersetzt. Dies bedeutet, dass wir nun folgende Konstituenten haben.

(12.10) [<Satz>[<DetP>[<Det>the␣][<Nomen>rat␣][<RelS>that␣drove␣a␣car␣]]
[<Präd>ran␣]]

Die terminalen Regeln haben keinen Effekt auf die gezeigte Konstituentenstruktur, weil die terminalen Elemente bereits eingesetzt sind. Der Schritt von Zeile 3 auf Zeile 4 ist somit wirkungslos. Dagegen wird beim Schritt von Zeile 4 auf Zeile 5 die Konstituente <RelS> in zwei Teile geteilt.

(12.11) [<Satz>[<DetP>[<Det>the␣][<Nomen>rat␣][<RelS>[<Comp>that␣]
[<Präd>drove␣a␣car␣]]]][<Präd>ran␣]]

Und so weiter. Am Ende bekommen wir folgende Struktur.

(12.12) [<Satz>[<DetP>[<Det>the␣][<Nomen>rat␣][<RelS>[<Comp>that␣]
[<Präd>[<TVerb>drove␣][<DetP>[<Det>a␣][<Nomen>car␣]]]]]
[<Präd>[<IVerb>ran␣]]]

Die andere Möglichkeit zur Visualisierung von (12.11) ist die Darstellung mittels eines Baumes. Diese geht intuitiv so: Jedes Teilwortvorkommen, das in (12.11) von einem Paar assoziierter Klammern eingeschlossen wird, heißt **Konstituente**. (Wir denken uns im Augenblick also die Klammern als Teile der Zeichenkette.) Die gesamte Zeichenkette ist also eine Konstituente ebenso wie

(12.13) $[_{\texttt{<DetP>}}[_{\texttt{<Det>}}\texttt{the\textvisiblespace}][_{\texttt{<Nomen>}}\texttt{rat\textvisiblespace}][_{\texttt{<RelS>}}[_{\texttt{<Comp>}}\texttt{that\textvisiblespace}]$
$[_{\texttt{<Präd>}}\texttt{drove\textvisiblespace a\textvisiblespace car\textvisiblespace}]]]$

oder auch

(12.14) $[_{\texttt{<Det>}}\texttt{the\textvisiblespace}]$

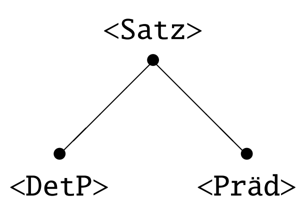
oder

(12.15) $[_{\texttt{<RelS>}}[_{\texttt{<Comp>}}\texttt{that\textvisiblespace}][_{\texttt{<Präd>}}\texttt{drove\textvisiblespace a\textvisiblespace car\textvisiblespace}]]$

Der Baum wird nun wie folgt konstruiert. Als Menge der sogenannten Knoten nehmen wir die (Vorkommen der) Konstituenten samt allen Vorkommen der einzelnen Worte. Auf dieser Menge, nennen wir sie K, erklären wir nun zwei Relationen, die Dominanz und die Präzedenz (oder “links-rechts-Abfolge”). Wir sagen, eine Konstituente γ dominiere eine Konstituente δ, falls δ echt in γ enthalten ist. Und γ sei **vor** δ, wenn sämtliche Teile von γ vor sämtlichen Teilen von δ stehen. Jede Konstituente bekommt noch eine Marke; ist sie ein Wort (ohne Klammern), so ist dies das Wort selbst. Hat sie die Form $[_{\alpha} \cdots]$, so ist ihre Marke α.

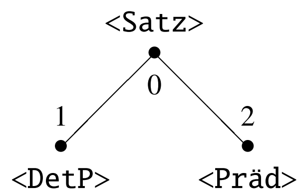
Beginnen wir mit der Konstruktion des Baumes. Die größte, alles umfassende Konstituente ist die gesamte Zeichenkette. Wir nennen sie auch die **Wurzel** des Baumes. Dies trägt notwendig die Marke `<Satz>`, denn das ist das Startsymbol. Die unmittelbar kleineren Konstituenten sind nun (12.13) sowie $[_{\texttt{<Präd>}}\texttt{ran\textvisiblespace}]]$. Ihre Marken sind `<DetP>` und `<Präd>`. Das erste steht links von dem zweiten. Dies ergibt folgendes Bild.

(12.16)



Das Bild enthält anstelle der Konstituenten (welche ja Vorkommen von Zeichenketten sind) nunmehr lediglich „Punkte”, an denen noch die Marke steht. Die Marken sind keine Bezeichnungen für die Konstituenten. Es kann sein, dass mehrere Knoten dieselbe Marke tragen. Deswegen werde ich zur besseren Übersicht in das Bild noch Zahlen für die Knoten einfügen.

(12.17)



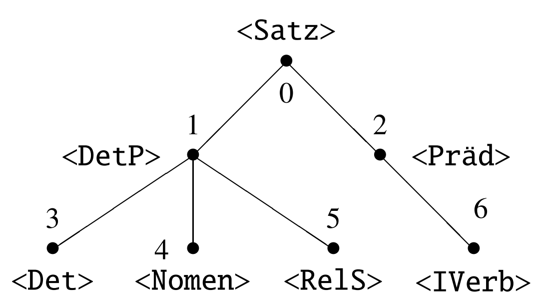
Das Bild gibt folgende Fakten wieder. Der Knoten 0 dominiert die Knoten 1 und 2. Und 1 ist links von 2. (Es sollte klar sein, dass 0 weder links von 1 oder 2 ist, noch dass 1 oder 2 links von 0 sind.)

Damit sind wir aber noch nicht fertig. Der Knoten 1 ist eine Konstituente, die ihrerseits aus drei Teilen besteht, nämlich

(12.18)
3 $[_{\langle\mathrm{Det}\rangle}\mathtt{the␣}]$
4 $[_{\langle\mathrm{Nomen}\rangle}\mathtt{rat␣}]$
5 $[_{\langle\mathrm{RelS}\rangle}[_{\langle\mathrm{Comp}\rangle}\mathtt{that␣}][_{\langle\mathrm{Präd}\rangle}\mathtt{drove␣a␣car␣}]]$

Gleichzeitig ist 2 eine Konstituente, die deckungsgleich mit der Konstituente `<IVerb>` ist (und die das Wort `/ran␣/` umfasst). Wir nehmen also noch einen weiteren Knoten hinzu (6), dessen Marke das Zeichen `<IVerb>` ist.

(12.19)



Es ist damit sowohl gesagt, wie die Dominanzbeziehungen sind (x dominiert y genau dann, wenn man von x aus den Linien entlang abwärts nach y gelangen kann) wie auch die Abfolge. Die Abfolge vererbt sich nach unten. Es ist 1 links von 2 und deswegen auch 3, 4 und 5 links von 2; weiter sind 3, 4, 5 links von 6 und 1 links von 6.

Um unsere Terminologie zu präzisieren, gebe ich hier ohne weitere Erklärung eine Definition.

Definition 12.3 (Syntaktischer Baum) *Ein* ***syntaktischer Baum*** *ist ein Quadrupel* $\langle K, D, L, \lambda \rangle$*, wo* K *eine endliche Menge ist, die Menge der* ***Knoten*** *des Baumes,* D *und* L *Relationen,* D *die* ***Dominanz*** *und* L *die* ***lineare Abfolge*** *und* $\lambda : K \to A \cup N$ *eine Funktion, die sogenannte* ***Markierungsfunktion****. Dabei gilt:*

① *D ist transitiv und irreflexiv.*

② *Ist $x \, D \, y$ und $z \, D \, y$, so ist $x = z$, $x \, D \, z$ oder $z \, D \, x$.*

③ *Es gibt ein $w \in K$ derart, dass für alle $x \neq w$: $w \, D \, x$. (Existenz einer Wurzel.)*

④ *L ist transitiv und irreflexiv.*

⑤ *L ist linear auf den Terminalknoten.*

⑥ *Genau dann ist $x \, L \, y$, wenn für alle u und v mit $x \, D \, u$ und $y \, D \, v$ gilt: $u \, L \, v$.*

Ein Knoten x heißt ***terminal****, wenn es kein y gibt mit xDy.*

Die Strukturzuweisung mit Hilfe der Ableitung ist aber leider nicht eindeutig. Hier ist ein Beispiel. Gegeben sei die Grammatik $\langle \{\mathtt{a}, \mathtt{b}\}, \{\mathtt{S}, \mathtt{A}, \mathtt{B}\}, \mathtt{S}, R \rangle$, wo R die folgenden Regeln enthält:

(12.20)
$$\begin{aligned} &\mathtt{S} \to \mathtt{AB} \\ &\mathtt{A} \to \mathtt{AS} \mid \mathtt{a} \\ &\mathtt{B} \to \mathtt{SB} \mid \mathtt{b} \end{aligned}$$

Schauen wir uns nun folgende Ableitung an.

(12.21) $\mathtt{S}, \mathtt{AB}, \mathtt{ASB}, \mathtt{AABB}, \mathtt{aABB}, \mathtt{aaBB}, \mathtt{aabB}, \mathtt{aabb}$

Ich behaupte, dass es zu dieser Ableitung *zwei* wesentlich verschiedene Strukturen gibt. Das Problem liegt hierbei in dem Schritt von /AB/ zu /ASB/. Wir können ihn vollziehen, indem wir A durch AS ersetzen; wir können ihn aber *auch* vollziehen, indem wir B durch SB ersetzen. Wir bekommen damit diese zwei Strukturen.

(12.22) $[_{\mathtt{S}}[_{\mathtt{A}}[_{\mathtt{A}}\mathtt{a}][_{\mathtt{S}}[_{\mathtt{A}}\mathtt{a}][_{\mathtt{B}}\mathtt{b}]]][_{\mathtt{B}}\mathtt{b}]]$, $\qquad [_{\mathtt{S}}[_{\mathtt{A}}\mathtt{a}][_{\mathtt{B}}[_{\mathtt{S}}[_{\mathtt{A}}\mathtt{a}][_{\mathtt{B}}\mathtt{b}]][_{\mathtt{B}}\mathtt{b}]]]$

Die erste besitzt eine Konstituente /`aab`/, die die zweite nicht hat; die zweite hate eine Konstituente, nämlich /`abb`/, die wiederum die erste nicht hat.

Damit nun die Ableitung eindeutig die Struktur rekonstruieren lässt, soll sie künftig nicht nur aus einer Folge von Zeichenketten bestehen. Sondern wir müssen auch noch hinzuschreiben, welches Nichtterminalzeichen jeweils ersetzt worden ist. Dies kann man am bequemsten durch Unterstreichung erreichen:

(12.23) $\underline{\mathtt{S}}, \underline{\mathtt{A}}\mathtt{B}, \mathtt{A}\underline{\mathtt{S}}\mathtt{B}, \underline{\mathtt{A}}\mathtt{ABB}, \mathtt{a}\underline{\mathtt{A}}\mathtt{BB}, \mathtt{aa}\underline{\mathtt{B}}\mathtt{B}, \mathtt{aab}\underline{\mathtt{B}}, \mathtt{aabb}$

Diese Ableitung passt eindeutig zu der ersten Struktur. Denn im zweiten Schritt wurde `A` ersetzt, und es lässt sich eindeutig sagen, welche Regel dabei verwendet wurde, wenn wir aus `A` im Ergebnis `ASB` bekommen. Es muss die Regel $\mathtt{A} \to \mathtt{AS}$ sein.

Die zweite Struktur wird nun durch die folgende Ableitung erzeugt.

(12.24) $\underline{\mathtt{S}}, \mathtt{A}\underline{\mathtt{B}}, \mathtt{A}\underline{\mathtt{S}}\mathtt{B}, \underline{\mathtt{A}}\mathtt{ABB}, \mathtt{a}\underline{\mathtt{A}}\mathtt{BB}, \mathtt{aa}\underline{\mathtt{B}}\mathtt{B}, \mathtt{aab}\underline{\mathtt{B}}, \mathtt{aabb}$

Ich gebe nun zum Schluss eine formale Version dieses Ableitungsbegriffs.

Definition 12.4 (Ableitung) *Es sei $\langle A, N, S, R\rangle$ eine kontextfreie Grammatik. Eine* ***Ableitung*** *einer Zeichenkette $\vec{\eta}$ ist eine Folge von Paaren $\langle C_i, \vec{\xi}_i\rangle$, $i < n+1$, wo für $i > 0$ $\vec{\xi}_i$ eine Zeichenkette über $A \cup N$ ist und C_i ein Vorkommen eines Nichtterminalzeichens X in $\vec{\xi}_{i-1}$ derart, dass es eine Regel $X \to \vec{\alpha} \in R$ gibt, sodass C_i ein Vorkommen von X in $\vec{\xi}_{i-1}$ benennt, dessen Ersetzung durch $\vec{\alpha}$ genau $\vec{\xi}_i$ ergibt. Für $i = 0$ gilt im Besonderen: C_0 ist beliebig, und $\vec{\xi}_0 = S$.*

Eine Ableitung beginnt also mit S und einem beliebigen Zahlenpaar. Anschließend gibt sie jeweils immer eine Zeichenkette an und ein Vorkommen, dessen Ersetzung zu dieser Zeichenkette aus der vorigen geführt hat. Die Benennung des Vorkommens hat uns bisher nicht interessiert, aber sie wird hier notwendig, weil sie für die Strukturzuweisung entscheidend ist.

Übungen

[51] Es sei folgende Grammatik zur Erzeugung von Deklarativ- und Imperativsätzen des Deutschen gegeben.

```
<CP>     → <DP><VP> | <VPimp>
<DP>     → <D><NP>
<NP>     → <N>
<VPimp>  → <Vimp>
<VP>     → <Vit> | <Vt><DP>
<D>      → der␣ | die␣ | das␣ | den␣
<N>      → Auto␣ | Lehrer␣ | Ärztin␣
<Vit>    → singt␣ | diskutiert␣
<Vt>     → singt␣ | sieht␣
<Vimp>   → singe␣ | diskutiere␣
```

Das Startsymbol ist <CP>. Leiten Sie einen Satz des Deutschen ab (mit mindestens 4 Worten). Zeichnen Sie zu dieser Ableitung die Struktur.

[52] Geben Sie zu der genannten Struktur eine weitere Ableitung an, die dieselbe Struktur hat.

[53] Leitet diese Grammatik auch ungrammatische Sätze ab? Wie kann man die Grammatik so verändern, dass dies ausgeschlossen ist?

[54] Wir werden dieser Grammatik auch Adverbien hinzufügen. Wir brauchen dazu (a) terminale Regeln (für jedes Wort mindestens eine) sowie (b) Regeln, die das Adverb richtig platzieren, damit (wenigstens) die folgenden Sätze ableitbar sind.

1. `der Lehrer singt laut`
2. `diskutiere heftig`
3. `die Ärztin sieht deutlich den Lehrer`

[55] Der geschilderte Fall der Grammatik **??** ist real. Betrachten wir den folgenden Satz.

(12.25)

`Der Polizist sagte, dass er den Mann mit einem Fernrohr gesehen hat.`

Wir können wahlweise /`den Mann mit einem Fernrohr`/ als Konstituente wählen oder /`mit einem Fernrohr gesehen hat`/. Im ersten Fall ist die PP /`mit einem Fernrohr`/ Adjunkt zu /`Mann`/, im zweiten Fall zu /`gesehen hat`/. Schreiben Sie eine Grammatik unf verifizieren Sie die Behauptungen.

Kapitel 13

Struktur und Bedeutung

Die Begriffe *Ableitung* und *Struktur* müssen wir scharf trennen. Ein und dieselbe Struktur kann mehrere Ableitungen besitzen. Dazu eine Überlegung. Angenommen, eine Zeichenkette besitze zwei Vorkommen von Nichtterminalzeichen.

(13.1) $\cdots X \cdots Y \cdots$

Wir können uns dann aussuchen, welches Vorkommen wir jetzt zuerst ersetzen. Angenommen, uns stehen die Regeln $\rho : X \to \vec{x}$ und $\sigma : Y \to \vec{y}$ zur Verfügung. Nach der Anwendung von ρ bekommen wir

(13.2) $\cdots \vec{x} \cdots Y \cdots$

Das Vorkommen von Y ist nach wie vor vorhanden. Man sieht leicht, dass es so lange vorhanden bleibt, wie wir darauf keine Regel anwenden. Das heißt, so lange wir Y nicht ersetzen, können wir jederzeit die Anwendung von σ vornehmen. Dies ist eine wichtige Vertauschungseigenschaft von kontextfreien Regeln. Aus ihr folgt, dass jeder Baum immer eine sogenannte linksseitige Ableitung besitzt. In einer linksseitigen Ableitung wird jeweils nur das Nichtterminalzeichen ersetzt, welches am weitesten links steht. (13.3) ist zum Beispiel eine linksseitige Ableitung. Man könnte aber auch das am weitesten recht stehende Symbol ersetzen. Dann bekommt man eine ganz andere Ableitung derselben Zeichenkette mit *demselben Strukturbaum.*

(13.3)

1. <Satz>
2. <DetP><Präd>
3. <DetP><IVerb>
4. <DetP>ran␣
5. <Det><Nomen><RelS>ran␣
6. <Det><Nomen><Comp><Präd>ran␣
7. <Det><Nomen><Comp><TVerb><DetP>ran␣
8. <Det><Nomen><Comp><TVerb><Det><Nomen>ran␣
9. <Det><Nomen><Comp><TVerb><Det>car␣ran␣
10. <Det><Nomen><Comp><TVerb>a␣car␣ran␣
11. <Det><Nomen><Comp>drove␣a␣car␣ran␣
12. <Det><Nomen>that␣drove␣a␣car␣ran␣
13. <Det>rat␣that␣drove␣a␣car␣ran␣
14. the␣rat␣that␣drove␣a␣car␣ran␣

Dies sind aber nicht die einzigen Ableitungen. Es gibt unzählige, die alle denselben Strukturbaum liefern.

Wenn wir nun Ableitungen haben, wieso sind dann Bäume so wichtig? Hierzu ein Beispiel.

(13.4)

<Satz> → <Satz><Konj><Satz> | <Sub><Präd>
<Konj> → und␣ | oder␣
<Sub> → Frank␣ | Ina␣ | Jens␣ | ⋯
<Präd> → liest␣ | kocht␣ | redet␣ | ⋯

Schauen wir uns folgenden Satz an:

(13.5) Jens␣liest␣und␣Ina␣redet␣oder␣Frank␣kocht␣

Dieser Satz besitzt sehr viele Ableitungen. Er besitzt auch zwei wesentlich verschiedene Bäume. Diese können wir so wiedergeben:

(13.6)

$[_{\langle\text{Satz}\rangle}[_{\langle\text{Satz}\rangle}[_{\langle\text{Satz}\rangle}[_{\langle\text{Sub}\rangle}\texttt{Jens␣}][_{\langle\text{Präd}\rangle}\texttt{liest␣}]]$
$[_{\langle\text{Konj}\rangle}\texttt{und␣}][_{\langle\text{Satz}\rangle}[_{\langle\text{Sub}\rangle}\texttt{Ina␣}][_{\langle\text{Präd}\rangle}\texttt{redet␣}]]]$
$[_{\langle\text{Konj}\rangle}\texttt{oder␣}][_{\langle\text{Satz}\rangle}[_{\langle\text{Sub}\rangle}\texttt{Frank␣}][_{\langle\text{Präd}\rangle}\texttt{kocht␣}]]]$

Unter Weglassen aller Klammern, die man ohnehin vorhersagen kann, bekommen wir

(13.7) [[Jens␣liest␣und␣Ina␣redet␣]oder␣[Frank␣kocht␣]]

Eine wesentlich andere Struktur ist

(13.8) `[[Jens␣liest␣]und␣[Ina␣redet␣oder␣Frank␣kocht␣]]`

In (13.8) ist /`Ina␣reded␣oder␣Frank␣kocht␣`/ eine Konstituente, in (13.7) dagegen nicht. Nehmen wir nun an, Frank kocht, Ina redet nicht, und Jens liest nicht. Dann ist (13.7) wahr: Denn Frank kocht. Aber (13.8) ist falsch: Denn Jens liest nicht.

Also ist nun ein und derselbe Satz wahr und falsch! Das ist so, weil der Aussagegehalt eines Satzes nicht eindeutig sein muss, zumindest nicht in natürlichen Sprachen. Wir sagen nun, dass die Bedeutung nicht einem Satz zugesprochen wird sondern der Satzstruktur, das heißt, seinem Strukturbaum. Oder auch anders: Während ein Satz mehrere Bedeutungen haben kann und so in einer gebenenen Situation sowohl wahr als auch falsch sein kann, ist dies für Strukturbäume nicht der Fall. Diese haben eine und nur eine Bedeutung, und somit sind sie in einer gegebenen Situation entweder wahr oder falsch, aber nicht beides. Die Bedeutung hängt also von der Struktur ab. Sie hängt aber *nicht* von der gewählten Ableitung dieser Struktur ab. Deswegen sagen wir, Bedeutung sei eine Eigenschaft von Strukturbäumen aber nicht von Ableitungen.

Solche Überlegungen gelten ganz allgemein auch für andere Typen von Konstituenten. Nehmen wir den folgenden Satz.

(13.9) `Xenakis war ein bedeutender Komponist und Architekt.`

(Yannis Xenakis, 1922 – 2001, war tatsächlich sowohl Komponist als auch Architekt und zumindest als Komponist auch ziemlich bedeutend.) Die Konstituente /`bedeutender Komponist und Architekt`/ lässt sich auf zwei Weisen verstehen. Die schwächere Interpretation ist, dass er zwar als Komponist bedeutend war aber nicht unbedingt als Architekt. Diesem entspricht die folgende Konstituentenstruktur. (Ich lasse unwesentliche Klammern weg.)

(13.10) `Xenakis war ein [[bedeutender Komponist] und Architekt].`

Die stärkere Lesart spricht ihm sowohl die Eigenschaft zu, bedeutender Komponist gewesen zu sein wie auch bedeutender Architekt. Diese Lesart hat die folgende Struktur.

(13.11) `Xenakis war ein [bedeutender [Komponist und Architekt]].`

Was kann der Grund sein für diese Tatsache? Ich erwähne hier nur folgendes Grundprinzip der Bedeutungstheorie, die sogenannte Kompositionalität.

> **Kompositionalität.** Die Bedeutung einer Konstituente hängt lediglich von der Bedeutung ihrer unmittelbaren Teilkonstituenten ab sowie der Art der Zusammensetzung.

Art der Zusammensetzung ist in dem Zusammenhang mit kontextfreien Grammatiken die Verkettung, bei der man also lediglich die Reihenfolge wählen kann, wenn die Konstituenten bereits vorliegen. Hat man also zwei Konstituenten $\vec{x}$ und $\vec{y}$, so gewinnt man daraus die Konstituenten $\vec{x} \cdot \vec{y}$ oder $\vec{y} \cdot \vec{x}$. Ferner gehört noch die syntaktische Kategorie dazu. Angewendet auf (13.10) bedeutet dies Folgendes. Die Bedeutung der Konstituente /`bedeutender Komponist und Architekt`/ bestimmt sich aus der Bedeutung von /`bedeutender Komponist`/ und der von /`Architekt`/. Sie ist in diesem Fall die Eigenschaft, beides zugleich zu sein, bedeutender Komponist und auch Architekt. Im Fall von (13.11) ist die Bedeutung eine Funktion der Bedeutung von /`bedeutend`/ und /`Komponist und Architekt`/; und dies ist die Eigenschaft, sowohl bedeutend als Komponist wie auch bedeutend als Architekt zu sein.

Kontextfreie Grammatiken lassen sich sehr stark vereinfachen.

Definition 13.1 (Normalform) *Ein kontextfreie Grammatik G ist in **Normalform**, wenn in einer Regel $A \to \vec{\gamma}$ die Zeichenkette $\vec{\gamma}$ entweder nur aus Terminalzeichen besteht oder nur aus Nichtterminalzeichen. G ist in **Chomskyscher Normalform**, wenn $\vec{\gamma}$ entweder nur aus Terminalzeichen besteht oder aus genau zwei Nichtterminalzeichen.*

Die Chomskysche Normalform beinhaltet, dass jeder Knoten in einem Baum, der nicht Terminalknoten ist, sich genau zweifach verzweigt. Des öfteren beschränkt man Terminalketten auf ein einzelnes Zeichen, aber das ist nicht hilfreich.

Wir formen eine beliebige Grammatik in Normalform um. Das geschieht wie folgt. Für jedes Terminalzeichen b sei N_b ein neues Nichtterminalzeichen. Falls die Grammatik eine Regel hat, die Nichtterminalzeichen und Terminalzeichen enthält, etwa $\mathtt{A} \to \mathtt{UbT}$, dann ersetzen wir die Terminalzeichen, hier /`b`/, durch die ihnen zugeordneten Nichtterminalzeichen, also $\mathtt{N_b}$, wir bekommen dann $\mathtt{A} \to \mathtt{UN_bT}$; schließlich fügen wir noch alle Regeln $N_b \to b$ hinzu. Damit ist Schritt zur Normalform getan. Nun können wir auch noch zur Chomskyschen Normalform übergehen.

Sei eine mehrstellige Regel gegeben, etwa $\mathtt{T} \to \mathtt{UVWX}$. Unter Zuhilfenahme neuer Nichtterminalsymbole $\mathtt{Y}$ und $\mathtt{Z}$ ersetzen wir diese Regel durch die Regeln:

$$\mathtt{T} \to \mathtt{UY}, \mathtt{Y} \to \mathtt{VZ}, \mathtt{Z} \to \mathtt{WX}. \tag{13.12}$$

Damit wird der einfache Übergang von $\vec{u}\mathtt{T}\vec{v}$ zu $\vec{u}\mathtt{UVWX}\vec{v}$ zu einer Serie

$$\vec{u}\mathtt{T}\vec{v}, \vec{u}\mathtt{UY}\vec{v}, \vec{u}\mathtt{UVZ}\vec{v}, \vec{u}\mathtt{UVWX}\vec{v} \tag{13.13}$$

Dieses Rezept lässt sich verallgemeinern.

Satz 13.2 (Pumplemma) *Es sei L eine kontextfreie Sprache. Dann existiert eine Zahl k derart, dass jedes Wort $\vec{x} \in L$ der Länge $\geq k$ zerlegt werden kann in $\vec{x} = \vec{u}\vec{v}\vec{w}\vec{y}\vec{z}$, sodass $\vec{v}\vec{y} \neq \varepsilon$ ist und für alle $i \in \mathbb{N}$ gilt $\vec{u}\vec{v}^i\vec{w}\vec{y}^i\vec{z} \in L$.*

Man beachte, dass i auch 0 oder 1 sein darf. Ist $i = 0$, so haben wir $\vec{u}\vec{w}\vec{z}$, ist $i = 1$, so bekommen wir $\vec{x}$. Für $i = 2$ bekommen wir $\vec{u}\vec{v}\vec{v}\vec{w}\vec{y}\vec{y}\vec{z}$. Mit dem Pumplemma lässt sich nicht zeigen, dass eine Sprache kontextfrei ist. Denn die Bedingung ist notwendig, aber nicht hinreichend. Es gibt Sprachen, die erfüllen die Pumpbedingung, ohne kontextfrei zu sein. Dazu gehört die Sprache $\{\vec{x}\mathsf{c}\vec{x} : \vec{x} \in (\mathsf{a}|\mathsf{b})^*\}$. Denn diese Sprache ist pumpbar. Auf der anderen Seite lässt sich mit dem Pumplemma zeigen, dass gewisse Sprachen *nicht* kontextfrei sind. So eine Sprache ist $\{\mathsf{a}^{n^2} : n \in \mathbb{N}\}$. Dazu nehmen wir ein beliebiges Wort a^{n^2} her und zerlegen es in fünf Teilworte. Diese haben die Länge $\leq n^2$, wobei zusätzlich $\vec{v}\vec{y}$ die Länge $g \leq n^2$ hat mit $g > 0$. Dann haben die Worte $\vec{u}\vec{v}^i\vec{w}\vec{y}^i\vec{z}$ die Länge $n^2 + (i-1)g$. Aber für beliebig viele i ist $n^2 + (i-1)g$ keine Quadratzahl.

Kommen wir nochmal auf die Sprache $C = \{\vec{x}\mathsf{c}\vec{x} : \vec{x} \in (\mathsf{a}|\mathsf{b})^*\}$ zurück. Zunächst einmal sagen wir, eine Sprache habe die **Pumpeigenschaft** oder sei **pumpbar**, wenn sie die in Satz 13.2 beschriebene Eigenschaft hat. Der Satz 13.2 besagt dann, dass eine kontextfreie Sprache pumpbar ist. Die Umkehrung gilt nicht.

Proposition 13.3 *Die Sprache C hat die Pumpeigenschaft, ist aber nicht kontextfrei.*

Beweis. Wir zeigen, dass C pumpbar ist mit $k = 3$. Sei $\vec{x}$ ein Wort der Länge mindestens 3. Dieses lässt sich wie folgt zerlegen: $\vec{x} = \vec{s}\mathsf{c}\vec{s}$. Ferner ist $\vec{s}$ nicht leer, also von der Form $a\vec{r}$ für einen Buchstaben a und ein (möglicherwiese leeres) Wort $\vec{r}$. Dann ist $\vec{x} = a\vec{r}\mathsf{c}a\vec{r}$. Setze $\vec{v} := a$, $\vec{y} := a$, $\vec{u} := \varepsilon$, $\vec{w} := \vec{r}\mathsf{c}$, sowie $\vec{z} := \vec{r}$. Dann ist $\vec{u}\vec{v}^k\vec{w}\vec{y}^k\vec{z} = a^k\vec{r}\mathsf{c}a^k\vec{r} \in C$.

Dass diese Sprache nicht kontextfrei ist, ist etwas schwierig zu sehen und soll hier nicht gezeigt werden. ⊣

Wie lässt sich nun die Pumpeigenschaft zeigen? Dazu greifen wir zu einem Trick. Der Einfachheit halber habe die Grammatik Chomsky Form. Jede Regel in unserer Grammatik, die die Form $A \to BC$ hat, wird in eine Regel der Form $A \to [_A BC]_A$, wo $[_A$ und $]_A$ zwei neue Terminalsymbole sind. Damit hat die neue Grammatik zwar nicht die Chomsky Form, aber das soll uns nicht stören. Wenn die neue Grammatik eine Zeichenkette erzeugt, so müssen wir lediglich die neuen Terminalzeichen auswischen, um eine Terminalkette der alten Grammatik zu erhalten. Nun schauen wir uns die Zeichenketten der neuen Grammatik an. Darin wimmelt es von Klammerpaaren der Form $\cdots [_A \cdots]_A \cdots$. Dabei schließen zwei Vorkommen eine Konstituente ein, falls die Anzahl der öffnenden Klammern vor dem Vorkommen von $[_A$ gleich der Anzahl der schließenden Vorkommen von Klammern nach $]_A$ ist. Jede Regelanwendung erzeugt ein solches Klammerpaar. Habe nun G n Nichtterminalsymbole. Ich behaupte, dass für große Zeichenketten immer die Form $\cdots [_A \cdots [_A \cdots]_A \cdots]_A \cdots$ haben für ein Nichtterminalsymbol A, wobei die äußeren und die inneren Klammerpaare jeweil seine Konstituente einschließen. Wir haben dann unsere Zerlegung $\vec{u}[_A \vec{v}[_A \vec{w}]_A \vec{y}]_A \vec{z}$.

Kapitel 14

Anhang: Axiomatische Mengenlehre

Die populärste Theorie der Mengen ist die sogenannte Zermelo-Fränkelsche Mengenlehre ZFC. (Der Buchstabe C steht hier für Englisch *choice*, Deutsch *Wahl*, weil das Auswahlaxiom mit dabei ist.) Ich will ihre Axiome hier kurz vorstellen. Die Sprache von ZFC kennt nur einen Typ von Objekt: den der Menge. Es gibt zwei Relationen zwischen Mengen, nämlich die Relation $\in$ und die Gleichheit, geschrieben $=$. Die Relation $\in$ ist weder transitiv noch reflexiv, sie ist sogar sehr kompliziert, wie die Axiome noch lehren werden. Ist $N \in M$, so sagt man, N sei **Element von** M. Dabei ist sowohl M als auch N eine Menge. Trotzdem unterscheidet man Menge und Element oft typographisch, etwa indem man schreibt $x \in N$. Wir sagen, M sei **Teilmenge von** N, in Zeichen $M \subseteq N$, falls für jedes $x \in M$ auch $x \in N$ ist. Die folgenden Postulate beschreiben die interne Struktur von Mengen.

> **Extensionalitätsaxiom.**
> Es gilt $M = N$ genau dann, wenn aus $x \in M$ folgt $x \in N$ und wenn aus $x \in N$ folgt $x \in M$.
> **Fundierungsaxiom.**
> Ist M eine Menge, so existiert keine Folge $M = M_0 \ni M_1 \ni M_2 \ni \cdots$.

Extensionalität lässt sich auch so formulieren: $M = N$ genau dann, wenn $M \subseteq N$ und $N \subseteq M$. Das Fundierungsaxiom versteht man am Besten, indem man sich Mengen als Schachteln vorstellt. Eine Schachtel kann wiederum Schachteln enthalten, und diese wiederum Schachteln, in denen wieder Schachteln sein können, und so weiter. Aber erstens ist eine Schachtel nicht in sich selbst enthalten, auch nicht in einer Schachtel, die in ihr enthalten ist, und zweitens kann man Schachteln nicht unendlich oft hintereinander auspacken. Die Schachteln, die man da auspackt, werden immer kleiner und kleiner, und irgendwann ist man fertig mit dem Auspacken. Die kleinste Schachtel ist notwendigerweise leer.

Nun folgen Postulate, die uns sagen, wie wir aus alten Mengen neue Menge machen können. Das erste ist das

> **Zweiermengenaxiom.**
> Zu je zwei Mengen M und N existiert die Menge $\{M, N\}$. Diese enthält M und N und nichts sonst.

Ist insbesondere $M = N$, so bekommen wir die Einermenge $\{M\}$. Es sei M eine Menge. Dann bezeichnen wir mit $\wp(M)$ die Menge der Teilmengen von M. Dass es sich hierbei tatsächlich um eine Menge handelt, sichert man sich durch ein Axiom. Dies formulieren wir so.

> **Potenzmengenaxiom.**
> Zu jeder Menge M existiert eine Menge, deren Elemente genau die Teilmengen von M sind. Wir bezeichnen diese Menge mit $\wp(M)$.

Ein nächstes Axiom sichert die Existenz der Vereinigung von Mengen. Wir wollen haben, dass mit Mengen N_1 und N_2 auch die Menge $N_1 \cup N_2$ existiert. Allerdings ist dies für viele Zwecke zu wenig. Man möchte gerne unendlich viele Mengen vereinigen können. Daher formuliert man das etwas kompliziertere

> **Vereinigungsmengenaxiom.**
> Zu jeder Menge M existiert eine Menge N derart, dass $x \in N$ genau dann, wenn ein $S \in M$ existiert mit $x \in S$. Wir bezeichnen N auch mit $\bigcup M$.

Die Idee hinter diesem Axiom ist die folgende. Ist $M = \{N_1, N_2\}$, so haben wir $\bigcup M = N_1 \cup N_2$. Da wir aber nicht nur zwei oder gar endlich viele Mengen vereinigen können wollen, sondern so viele wie möglich, fassen wir die zu vereinigenden Mengen in eine Menge M zusammen und bilden anschließend $\bigcup M$. Diese enthält genau die Elemente der zu vereinigenden Mengen.

Ist $\varphi(x)$ eine Eigenschaft von Mengen, so schreibt man $\{x : \varphi(x)\}$ für die Menge aller Mengen, welche $\varphi(x)$ erfüllen. Allerdings dürfen wir nicht unkritisch damit umgehen, wie folgendes Beispiel lehrt.

Satz 14.1 (Russell) *Es gibt keine Menge* $\{x : x \notin x\}$.

Beweis. Angenommen, es gibt diese Menge. Nennen wir sie V. Dann ist entweder (Fall 1) $V \in V$ oder (Fall 2) $V \notin V$. Im Fall 1 gilt $V \notin V$, nach Definition von V. Im Fall 2 gilt $V \in V$, wiederum nach Definition von V. Ein Widerspruch. ⊣

Ich merke an, dass für jede Menge M gilt: $M \notin M$. Also ist die Menge, wenn sie denn existiert, nicht leer, und sie enthält jede nur denkbare Menge. Man nennt daher V auch das **Universum**. Das Universum können wir uns zwar irgendwie vorstellen, aber es ist keine Menge. Um dennoch nach solch einem Prinzip sorgenfrei Mengen schaffen zu können, definiert man wie folgt.

> **Aussonderungsaxiom.**
> Es sei $\varphi(x)$ eine Eigenschaft von Mengen und sei M eine Menge. Dann ist $\{x : x \in M \text{ und } \varphi(x)\}$ eine Menge. Diese enthält genau diejenigen x, die in N und die φ erfüllen.

Zu je zwei Mengen M und N existieren ferner die Mengen $M \cap N$, $M \cup N$, $M - N$. Die erste und die dritte existieren nach dem Aussonderungsaxiom (wir sondern aus M alle die Elemente aus, die zu N (bzw. nicht zu N) gehören), die zweite Menge existiert nach dem Vereinigungsmengenaxiom.

Am schwierigsten zu begreifen ist vielleicht die Intention des sogenannten *Auswahlaxioms*.

> **Auswahlaxiom.**
> Es sei M eine Menge. Dann existiert eine sogenannte Auswahlfunktion f, welche zu jeder nichtleeren Menge $N \in M$ ein Element $f(N) \in N$ benennt.

Man beachte, dass die Auswahlfunktion eine Menge ist. Als Funktion eingesetzt erlaubt sie uns, aus jedem Mengensystem gleichzeitig in allen darin enthaltenen Mengen ein und nur ein Element auszuwählen.

Teil II

Formale Methoden 2: Logik

Kapitel 15

Einführung: Syntax und Semantik

Ich beginne mit ein paar allgemeinen Bemerkungen zu Sprachen. Sprachen sind spezielle Zeichensysteme. Ein **Zeichen** ist ein Paar $\langle e, m\rangle$, wo e die **Zeichensubstanz** oder der **Ausdrucksträger** ist und m die **Bedeutung**. Ein Zeichensystem ist zum Beispiel das System $\mathcal{V}$ der Verkehrsschilder in Deutschland. Darin finden wir zum Beispiel das Stopschild. Seine Bedeutung können wir einfach mit *Halt!* wiedergeben. Dabei ist weder das Schild selbst noch seine Bedeutung in $\mathcal{V}$, sondern nur das Zeichen, das heißt, das Paar $\sigma = \langle$🛑$, \textit{Halt!}\rangle$. Der Bedeutungsträger von σ ist also 🛑, die Bedeutung von σ ist *Halt!*. Im normalen Sprachgebrauch sagt man allerdings, es habe das Stopschild 🛑 die Bedeutung *Halt!* und nicht das Zeichen, welches ja das Paar $\langle$🛑$, \textit{Halt!}\rangle$ ist. Aber das ist lediglich eine geduldete Nachlässigkeit. Bedeutung kommt eigentlich nur dem Zeichen zu, nicht dem Ausdrucksträger; alternativ lässt sich sagen, der Träger habe Bedeutung in einem Zeichensystem. Das ist wichtig, weil es die Tatsache unterstreicht, dass Bedeutungen dem Träger nicht inhärent zugeordnet sind, sondern ganz verschieden vereinbart werden, je nach Zeichensystem. Dass etwas ein überhaupt Bedeutungsträger ist und welche Bedeutung es hat, wird von dem Zeichensystem entschieden. Wir definieren nun wie folgt.

Definition 15.1 *Ein* ***Zeichensystem*** *ist eine Menge von Zeichen.*

Da nun ein Zeichen ein Paar ist, können wir auch sagen, ein Zeichensystem ist ein Relation. Ist dann E eine Menge von Ausdrucksträgern und M eine Menge von Bedeutungen, so ist $\mathcal{Z} \subseteq E \times M$ ein Zeichensystem über E und M. Für Sprachen gilt dann, dass $E = A^*$ für ein gewisses A ist. An dieser Stelle sei gesagt, dass die Mengen E und M frei gewählt werden können.

Die Menge $\mathcal{Z}$ darf endlich oder unendlich sein. Das Zeichensystem der Verkehrsschilder in Deutschland ist endlich (zumindest, was die offiziell anerkannten Verkehrsschilder angeht). In diesem Zeichensystem finden wir nun unser Paar ⟨, *Halt!*⟩, es ist schlicht ein Element von $\mathcal{V}$. Die Zuschreibung von Bedeutung zu einem Träger ist relativ zu dem System, man kann die Bedeutung willkürlich und unterschiedlich festlegen, und das geschieht auch. Zeichensysteme sind schlicht konventionell gefasst. So bedeutet `/tak/` im Schwedischen *Danke!* und im Polnischen *ja*. Und dies ist so, weil sich das Schwedische auf eine Weise entwickelt hat, das Polnische auf eine andere; und jede Sprache ist frei, irgendwelche Zeichenketten als Ausdrucksträger zu wählen und ihnen irgendeine Bedeutung zuzuschreiben. Das bedeutet wiederum, dass auf der einen Seite nichts per se ein Ausdrucksträger ist. Sondern ein Gegenstand g wird zum Ausdrucksträger dadurch, dass er in einem Zeichen $\langle g, m\rangle$ des Systems $\mathcal{V}$ auftritt.

Definition 15.2 *Ist $\mathcal{Z} \subseteq E \times M$, so hat $e \in E$ die* ***Bedeutung*** *m in $\mathcal{Z}$, falls $\langle e, m\rangle \in \mathcal{Z}$. m wird in diesem Fall durch e in $\mathcal{Z}$* ***bezeichnet****.*

Alles und jedes kann im Prinzip Ausdrucksträger in irgendeinem Zeichensystem sein. Wir bevorzugen natürlich sinnliche Dinge, also solche, die wir wahrnehmen können: Bilder, Buchstaben, Laute, und so weiter. Wichtig ist auch, dass es keinerlei weitere Beschränkungen gibt.

- Zu einem gegebenen e kann es beliebig viele m geben mit $\langle e, m\rangle \in \mathcal{Z}$.
- Zu einem gegeben m kann es beliebig viele e geben mit $\langle e, m\rangle \in \mathcal{Z}$.

Insbesondere ist es möglich, dass es zu einem e *kein* m gibt mit $\langle e, m\rangle \in \mathcal{Z}$ (e ist nennt man dann auch *sinn-* oder *bedeutungslos*), und dass es mehrere m gibt (e heißt dann *mehrdeutig*). Und so ist es möglich, dass es zu einem m *kein* e gibt mit $\langle e, m\rangle \in \mathcal{Z}$, oder eben mehrere. Im Regelfall gibt es in der Logik sogar unendlich viele e mit $\langle e, m\rangle \in \mathcal{Z}$.

Wenn wir nun vereinbaren, das Zeichensystem der Verkehrsschilder enthalte das Paar ⟨, *Halt!*⟩, dann ist das (in der Regel) abstrakt gemeint. Denn das Symbol  darf fortan an einem beliebigen Ort stehen und bekommt seine Bedeutung kraft des Zeichensystems. Mit anderen Worten, wir unterscheiden zwischen einem **Typ** und seinem **Vorkommen** (oder auch zwischen **Schema** und **Ausprägung**). In dem Zeichensystem ist lediglich der Typ „Stopschild" festgelegt, dort findet man keine konkreten Stopschilder. Und es wird dem Typ „Stopschild" die Bedeutung *Halt!* gegeben. Jedes Schild, das dem Schema genügt wird damit zu einem

Vorkommen ebendieses Typs und bekommt jetzt die Bedeutung *Halt!*. Mit Bedeutungen geht es ebenso. *Halt!* ist auch nur abstrakt. Es sagt nicht, wer spricht, wer angesprochen ist und wann. Wer soll also wo halten? Das ist so geregelt: halten muss der, der bei einem konkreten Stopschild ankommt. Und er muss in ebendiesem Moment anhalten und eben dort, wo dieses Stopschild sich befindet. Der Befehl geht in diesem Fall von dem Gesetzgeber aus. Viele Einzelheiten müssen hierbei geregelt werden. Nicht jedes Schild, das dem Stopschild-Schema genügt, entfaltet seine Wirkung. Wenn es zum Beispiel auf einem Laster aufgeladen ist, so muss niemand anhalten, der es zufällig sieht. Insofern gibt es noch einen feinen Unterscheid zwischen der Bedeutung (die dieses konkrete Schild durchaus hat) und der Wirkkraft, die es entfaltet. Wer dies Schild sieht, weiß sicher, was es bedeutet. Aber er wird nicht anhalten, weil festgelegt ist, dass er nicht halten muss, wenn das Schild nicht an einem bestimmten Ort auf bestimmte Weise befestigt ist. In dieser Vorlesung werde ich nicht weiter über die Wirkkraft sprechen. Es soll hier nur um die Frage gehen, was die Bedeutung von bestimmten Zeichen ist, unabhängig von Fragen nach der Wirkung.

Im Fall von endlichen Zeichensystemen kann man eine Liste anfertigen, die sämtliche Zeichen enthält. Für unendliche Zeichensysteme ist dies schlicht unmöglich. In diesem Fall muss man ein Erzeugungssystem, genannt **Grammatik**, angeben. Eine Grammatik sagt uns, wie wir aus bereits bekannten Zeichen neue machen können. Ich gebe dazu ein Beispiel. Unser Zeichensystem $\mathcal{B}$ bestehe aus den Paaren $\langle z, n\rangle$, wo z eine Folge aus /$\mathtt{L}$/ und /$\mathtt{0}$/ ist und n die davon repräsentierte Binärzahl.

$$\mathcal{B} = \{\langle \mathtt{0}, 0\rangle, \langle \mathtt{L}, 1\rangle, \langle \mathtt{00}, 0\rangle, \langle \mathtt{0L}, 1\rangle, \langle \mathtt{L0}, 2\rangle, \langle \mathtt{LL}, 3\rangle, \cdots\} \tag{15.1}$$

Dieses System ist unendlich. Es enthält zum Beispiel das Paar $\langle \mathtt{L0LL00L}, 89\rangle$. Dies weiß ich aber nicht deshalb, weil ich dies auswendig gelernt habe, sondern weil ich ein Prinzip kenne, wie man sich alle Paare erzeugen kann. Dies läuft so. Die Paare $\langle \mathtt{0}, 0\rangle$ und $\langle \mathtt{L}, 1\rangle$ muss man kennen. Sind dann $\langle \vec{x}, n\rangle$ bereits in der Menge, so sind die folgenden Paare auch in der Menge:

$$\langle \vec{x}\cdot\mathtt{0}, 2n\rangle \qquad \langle \vec{x}\cdot\mathtt{L}, 2n+1\rangle \tag{15.2}$$

Oder formal:

$$\begin{aligned}&(\forall\langle \vec{x}, n\rangle)(\langle \vec{x}, n\rangle \in \mathcal{B} \to \langle \vec{x}\cdot\mathtt{0}, 2n\rangle \in \mathcal{B})\\&(\forall\langle \vec{x}, n\rangle)(\langle \vec{x}, n\rangle \in \mathcal{B} \to \langle \vec{x}\cdot\mathtt{L}, 2n+1\rangle \in \mathcal{B})\end{aligned} \tag{15.3}$$

Als drittes muss man noch sagen, dass nichts sonst in der Menge ist, das heißt, dass dieses Schema alle Zeichen erzeugt.

Probieren wir dies aus. Meine Zeichenkette ist $\mathtt{LOLLOOL}$. Ich erzeuge sie wie folgt. Ich beginne mit $\langle \mathtt{L}, 1\rangle$. Daraus bilde ich $\langle \mathtt{L}\cdot\mathtt{0}, 2\cdot 1\rangle = \langle \mathtt{L0}, 2\rangle$. Daraus bilde ich $\langle \mathtt{L0}\cdot\mathtt{L}, 2\cdot 2+1\rangle = \langle \mathtt{L0L}, 5\rangle$, und so weiter.

$$\begin{array}{lr} \langle \mathtt{L}, & 1\rangle \\ \langle \mathtt{L0}, & 2\rangle \\ \langle \mathtt{L0L}, & 5\rangle \\ \langle \mathtt{L0LL}, & 11\rangle \\ \langle \mathtt{L0LL0}, & 22\rangle \\ \langle \mathtt{L0LL00}, & 44\rangle \\ \langle \mathtt{L0LL00L}, & 89\rangle \end{array} \tag{15.4}$$

Dies bedeutet nun, dass wir sagen, die Exponenten unserer Sprache sind in irgendeiner Weise zusammengebaut worden. Die vorliegende Grammatik sagt, der nächste Buchstabe wird an das Wortende angehängt; die Zeichenkette wird von links nach rechts erzeugt, und nicht etwa von rechts nach links. Das ist nicht so trivial, wie es sich anhört. Es ist nämlich ganz und gar unmöglich, einen Algorithmus anzugeben, der die Bedeutung von $\mathtt{L}\vec{x}$ aus der Bedeutung von $\vec{x}$ zu berechnen; denn man beachte, dass mit $\vec{x}$ auch $\mathtt{0}\vec{x}$, $\mathtt{00}\vec{x}$, $\mathtt{000}\vec{x}$ sämtliche dieselbe Zahl darstellen, aber $\mathtt{L}\vec{x}$, $\mathtt{L0}\vec{x}$, $\mathtt{L00}\vec{x}$, $\mathtt{L000}\vec{x}$ allesamt verschiedene Zahlen sind!

Man nennt die Lehre der Zusammensetzung von Ausdrucksträgern auch **Syntax**, die Lehre von der Zusammensetzung ihrer Bedeutung **Semantik**.

Kapitel 16

Aussagen

In diesem Kapitel werden wir Aussagen studieren. Der Sinn und Zweck ist es, einerseits zu verdeutlichen, wie man feststellen kann, ob eine Aussage nun wahr oder falsch ist, andererseits aber zu zeigen, wie Eigenschaften der Repräsentation wichtig sind für das Argumentieren mit Aussagen.

Im Sprachgebrauch sind Aussagen gewisse Sätze. Aussagen sind Sätze, die wahr sein können, genauer, *Sätze, bei denen es sinnvoll ist zu sagen, dass sie wahr sind*. So ist zum Beispiel /`Sokrates war Schuster.`/ eine Aussage, weil sie wahr ist. /`Napoleon war Chinese.`/ ist auch eine Aussage, sie ist nicht wahr, aber wenigstens ist es sinnvoll zu sagen, dass sie wahr ist. (Auch eine falsche Aussage ist sinnvoll.) /`Morgen wird es in Bielefeld regnen.`/ ist eine Aussage, weil sie wahr sein kann und auch falsch. Im Augenblick ist sie weder wahr noch falsch, oder jedenfalls wissen wir nicht, ob sie nun wahr ist oder nicht. Es ist in jedem Fall aber sinnvoll zu sagen, sie sei wahr.

In der folgenden Definition werden auf Kapitel 6 zurückgreifen. Aussagen sind gewisse Zeichenketten, und die Sprache der Aussagenlogik dementsprechend eine Menge von Zeichenketten.

Definition 16.1 (Syntax) *Es sei* $A = \{\mathtt{w}, \mathtt{f}, \mathtt{(}, \mathtt{)}, \wedge, \vee, \neg, \rightarrow\}$. $\vec{x} \in A^*$ *ist eine* ***Aussage****, falls eine der folgenden Behauptungen zutrifft:*

① $\vec{x} = \mathtt{w}$,

② $\vec{x} = \mathtt{f}$,

③ $\vec{x} = \mathtt{(}\neg\vec{y}\mathtt{)}$ *für eine Aussage* $\vec{y}$,

④ $\vec{x} = \mathtt{(}\vec{y}\wedge\vec{z}\mathtt{)}$ *für Aussagen* $\vec{y}$ *und* $\vec{z}$, ⑤ $\vec{x} = (\vec{y}\lor\vec{z})$ *für Aussagen* $\vec{y}$ *und* $\vec{z}$,

⑥ $\vec{x} = (\vec{y}\to\vec{z})$ *für Aussagen* $\vec{y}$ *und* $\vec{z}$.

Es sind also folgende Zeichenketten Aussagen:

$$\texttt{w}, \texttt{(}\neg\texttt{f)}, \texttt{(w}\land\texttt{f)}, \texttt{(w}\to\texttt{(w}\land\texttt{(}\neg\texttt{f)))} \tag{16.1}$$

Diese Definition ist nur scheinbar zirkulär; wir sagen zum Beispiel, dass $/\texttt{(}\neg\vec{y}\texttt{)}/$ eine Aussage ist, weil $\vec{y}$ eine Aussage ist. Nun hat $\vec{y}$ aber kleinere Länge als $/\texttt{(}\neg\vec{y}\texttt{)}/$. Mit jeder Anwendung der Klauseln ③ bis ⑥ reduzieren wir das Problem auf kleinere Zeichenketten, bis wir schließlich bei unzerlegbaren Zeichenketten ankommen. Dann greift ① oder ② und sagt uns, dass nur /`w`/ bzw. /`f`/ Aussagen sind.

Die Zeichen ¬, ∧, ∨ und → heißen auch **Operationszeichen**. Wir nennen ∧ die **Konjunktion**, ∨ die **Disjunktion** und → die **Implikation**. Noch ein Wort zur Notation. Ich benutze sogenannte Schreibmaschinenschrift für Zeichen, wie sie auf dem Papier zu sehen sind. Weil ich aber manche Zeichenketten nur schematisch angeben kann, sind dort auch Variable für Zeichenketten zu sehen, wie etwa $\vec{x}$, $\vec{y}$. Alles, was nicht in Schreibmaschinenschrift erscheint, ist also formal nicht als Teil der echten Zeichenkette zu sehen sondern ist nur ein Ersatz. Es ist ein Statthalter für das, was wirklich dasteht.

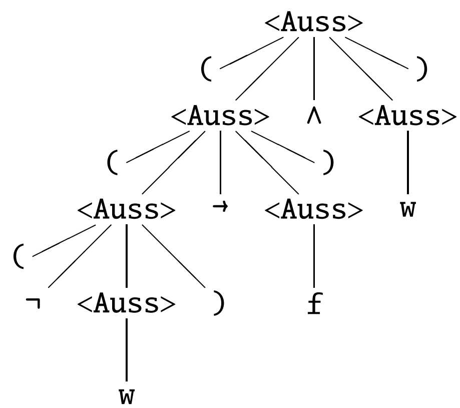
Die obige Definition lässt sich mit Hilfe einer Grammatik wie folgt vereinfachen.

$$\begin{aligned} \texttt{<Auss>} \to\ & \texttt{w} \mid \texttt{f} \mid \texttt{(}\neg\texttt{<Auss>)} \mid \texttt{(<Auss>}\land\texttt{<Auss>)} \\ & \mid \texttt{(<Auss>}\lor\texttt{<Auss>)} \mid \texttt{(<Auss>}\to\texttt{<Auss>)} \end{aligned} \tag{16.2}$$

Diese Grammatik erlaubt uns, Aussagen einen Strukturbaum zuzuordnen. Die geschieht wie in Kapitel 12 beschrieben. Zum Beispiel hat die Formel $\texttt{(((}\neg\texttt{w)}\to\texttt{f)}\land\texttt{w)}$

den folgenden Baum.

(16.3)



Einer Aussage werden wir nun einen Wahrheitswert zuordnen. Dies wird in zwei Schritten geschehen. Zunächst werde ich wie im vorigen Kapitel beschrieben Paare $\langle e, m\rangle$ erzeugen, wo e eine Aussage und m ein Wahrheitswert ist; und dann werde ich zeigen, dass m eindeutig von e bestimmt ist. Damit ist es offiziell erlaubt, von *dem* Wahrheitswert einer Aussage zu sprechen.

Definition 16.2 (Semantik) *Die Menge* $T := \{0, 1\}$ *ist die Menge der Wahrheitswerte. Es sei U die kleinste Menge, die Folgendes erfüllt.*

① $\langle \mathtt{w}, 1\rangle \in U$,

② $\langle \mathtt{f}, 0\rangle \in U$.

③ *Für die Negation:*

- *falls* $\langle \vec{x}, 1\rangle \in U$, *so* $\langle \mathtt{(}\neg\vec{x}\mathtt{)}, 0\rangle \in U$;
- *falls* $\langle \vec{x}, 0\rangle \in U$, *so* $\langle \mathtt{(}\neg\vec{x}\mathtt{)}, 1\rangle \in U$.

④ *Für die Konjunktion:*

- *falls* $\langle \vec{x}, 1\rangle \in U$ *und* $\langle \vec{y}, 1\rangle \in U$, *so* $\langle \mathtt{(}\vec{x}\wedge\vec{y}\mathtt{)}, 1\rangle \in U$;
- *falls* $\langle \vec{x}, 1\rangle \in U$ *und* $\langle \vec{y}, 0\rangle \in U$, *so* $\langle \mathtt{(}\vec{x}\wedge\vec{y}\mathtt{)}, 0\rangle \in U$;
- *falls* $\langle \vec{x}, 0\rangle \in U$ *und* $\langle \vec{y}, 1\rangle \in U$, *so* $\langle \mathtt{(}\vec{x}\wedge\vec{y}\mathtt{)}, 0\rangle \in U$;
- *falls* $\langle \vec{x}, 0\rangle \in U$ *und* $\langle \vec{y}, 0\rangle \in U$, *so* $\langle \mathtt{(}\vec{x}\wedge\vec{y}\mathtt{)}, 0\rangle \in U$.

⑤ *Für die Disjunktion:*

- *falls* $\langle\vec{x}, 1\rangle \in U$ *und* $\langle\vec{y}, 1\rangle \in U$, *so* $\langle\texttt{(}\vec{x}\texttt{∨}\vec{y}\texttt{)}, 1\rangle \in U$;
- *falls* $\langle\vec{x}, 1\rangle \in U$ *und* $\langle\vec{y}, 0\rangle \in U$, *so* $\langle\texttt{(}\vec{x}\texttt{∨}\vec{y}\texttt{)}, 1\rangle \in U$;
- *falls* $\langle\vec{x}, 0\rangle \in U$ *und* $\langle\vec{y}, 1\rangle \in U$, *so* $\langle\texttt{(}\vec{x}\texttt{∨}\vec{y}\texttt{)}, 1\rangle \in U$;
- *falls* $\langle\vec{x}, 0\rangle \in U$ *und* $\langle\vec{y}, 0\rangle \in U$, *so* $\langle\texttt{(}\vec{x}\texttt{∨}\vec{y}\texttt{)}, 0\rangle \in U$.

⑥ *Für die Implikation:*

- *falls* $\langle\vec{x}, 1\rangle \in U$ *und* $\langle\vec{y}, 1\rangle \in U$, *so* $\langle\texttt{(}\vec{x}\texttt{→}\vec{y}\texttt{)}, 1\rangle \in U$;
- *falls* $\langle\vec{x}, 1\rangle \in U$ *und* $\langle\vec{y}, 0\rangle \in U$, *so* $\langle\texttt{(}\vec{x}\texttt{→}\vec{y}\texttt{)}, 0\rangle \in U$;
- *falls* $\langle\vec{x}, 0\rangle \in U$ *und* $\langle\vec{y}, 1\rangle \in U$, *so* $\langle\texttt{(}\vec{x}\texttt{→}\vec{y}\texttt{)}, 1\rangle \in U$;
- *falls* $\langle\vec{x}, 0\rangle \in U$ *und* $\langle\vec{y}, 0\rangle \in U$, *so* $\langle\texttt{(}\vec{x}\texttt{→}\vec{y}\texttt{)}, 1\rangle \in U$.

Ist $\langle\vec{x}, 1\rangle \in U$, *so heißt* $\vec{x}$ *eine* ***wahre Aussage****. Falls* $\langle\vec{x}, 0\rangle \in U$, *so heißt* $\vec{x}$ *eine* ***falsche Aussage****.*

Im Prinzip könnte eine gegebene Aussage $\vec{x}$ sowohl wahr als auch falsch sein. Dazu müsste nur sowohl $\langle\vec{x}, 1\rangle \in U$ wie auch $\langle\vec{x}, 0\rangle \in U$ sein. Wir werden weiter unten sehen, dass dies allerdings in dieser Sprache ausgeschlossen ist.

Ich gebe nun ein Beispiel. Wir wollen wissen, ob `(((¬w)∨f)∨(w→f))` wahr oder falsch ist. Dazu überlegen wir wie folgt. Es ist $\langle\texttt{w}, 1\rangle \in U$ (`w` ist wahr) und deswegen $\langle\texttt{(¬w)}, 0\rangle \in U$. Es ist auch $\langle\texttt{f}, 0\rangle \in U$. Daraus folgt nun $\langle\texttt{((¬w)∨f)}, 0\rangle \in U$. Es folgt $\langle\texttt{(w→f)}, 0\rangle \in U$, und damit $\langle\texttt{(((¬w)∨f)∨(w→f))}, 0\rangle \in U$. Die Aussage ist somit falsch. (Wie wir noch sehen werden, ist sie deswegen nicht wahr, was keineswegs banal ist.) Dies kann man auch in eine Tabelle bringen.

(16.4)

Zeile		Zeichen
1	(①)	$\langle\texttt{w}, 1\rangle$
2	(③ : 1)	$\langle\texttt{(¬w)}, 0\rangle$
3	(②)	$\langle\texttt{f}, 0\rangle$
4	(⑤ : 2, 3)	$\langle\texttt{((¬w)∨f)}, 0\rangle$
5	(⑥ : 3, 1)	$\langle\texttt{(f→w)}, 0\rangle$
6	(⑤ : 4, 5)	$\langle\texttt{(((¬w)∨f)∨(w→f))}, 0\rangle$

Hierbei steht rechts das abgeleitete Zeichen, links die Nummer der Zeile und danach ein Grund, warum das Zeichen als abgeleitet gelten darf. Entweder es ist nach ① oder ② in U, oder es kann aus abgeleiteten Zeichen mittels ③ – ⑥ hergestellt werden. Die Zeilennummer ist dann nach dem Doppelpunkt vermerkt. Die formale Struktur einer solchen Ableitung ist für uns hier nicht wesentlich. Sie soll nur verdeutlichen, dass man jeden Schritt genau dokumentieren und damit nachvollziehbar machen kann.

Die Definition 16.2 lässt sich wesentlich kompakter darstellen. Dazu bedienen wir uns der folgenden Funktionen:

$$\begin{array}{c|c} & - \\ \hline 0 & 1 \\ 1 & 0 \end{array} \qquad \begin{array}{c|cc} \cap & 0 & 1 \\ \hline 0 & 0 & 0 \\ 1 & 0 & 1 \end{array} \qquad \begin{array}{c|cc} \cup & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 1 & 1 \end{array} \qquad \begin{array}{c|cc} \to & 0 & 1 \\ \hline 0 & 1 & 1 \\ 1 & 0 & 1 \end{array} \tag{16.5}$$

Damit können wir die Definition so aufschreiben:

Definition 16.3 *Die Menge $T := \{0, 1\}$ ist die Menge der Wahrheitswerte. Es sei U die kleinste Menge, die Folgendes erfüllt.*

① $\langle \mathtt{w}, 1\rangle \in U$.

② $\langle \mathtt{f}, 0\rangle \in U$.

③ *Falls* $\langle \vec{x}, p\rangle \in U$, *so* $\langle \mathtt{(\neg}\vec{x}\mathtt{)}, -p\rangle \in U$.

④ *Falls* $\langle \vec{x}, p\rangle \in U$ *und* $\langle \vec{y}, q\rangle \in U$, *so* $\langle \mathtt{(}\vec{x}\mathtt{\wedge}\vec{y}\mathtt{)}, p \cap q\rangle \in U$.

⑤ *Falls* $\langle \vec{x}, p\rangle \in U$ *und* $\langle \vec{y}, q\rangle \in U$, *so* $\langle \mathtt{(}\vec{x}\mathtt{\vee}\vec{y}\mathtt{)}, p \cup q\rangle \in U$.

⑥ *Falls* $\langle \vec{x}, p\rangle \in U$ *und* $\langle \vec{y}, q\rangle \in U$, *so* $\langle \mathtt{(}\vec{x}\mathtt{\to}\vec{y}\mathtt{)}, p \to q\rangle \in U$.

Man beachte, dass die Definition 16.3 dasselbe Zeichensystem definiert wie die Definition 16.2. Dies lässt sich leicht verifizieren.

Satz 16.4 (Eindeutige Lesbarkeit) *Es sei $\vec{x}$ eine Aussage. Dann tritt genau einer der folgenden Fälle ein:*

① $\vec{x} = \mathtt{w}$.

② $\vec{x} = \mathtt{f}$.

③ $\vec{x} = \mathtt{(\neg}\vec{y}\mathtt{)}$.

④ $\vec{x} = \mathtt{(}\vec{y}\mathtt{\wedge}\vec{z}\mathtt{)}$.

⑤ $\vec{x} = (\vec{y}\vee\vec{z})$.

⑥ $\vec{x} = (\vec{y}\to\vec{z})$.

Im Fall ③ ist $\vec{y}$ eindeutig bestimmt; in den Fällen ④ – ⑥ sind $\vec{y}$ und $\vec{z}$ jeweils eindeutig bestimmt.

Bevor ich dies beweise, gehe ich noch einmal die Formulierung und die Folgerungen daraus durch. Zunächst einmal heiße eine Aussage **einfach**, falls sie die Form /`w`/ oder /`f`/ hat. Eine Aussage, die nicht einfach ist, heißt **komplex**. Eine komplexe Aussage hat genau eine der Formen ③ – ⑥. Im Fall ③ heißt ¬ das **Hauptsymbol**, im Fall ④ – ⑥ ist das Hauptsymbol ∧, ∨ bzw. →. Die Aussagen $\vec{y}$ und $\vec{z}$ heißen dann die unmittelbaren Teilaussagen von $\vec{x}$. (Um genau zu sein, müssten wir von Vorkommen des Hauptsymbols sprchene, aber das würde die Sprache unnötig verkomplizieren.) Der Satz sagt nun, dass für eine komplexe Aussage das Hauptsymbol sowie seine unmittelbaren Teilaussagen eindeutig sind. Dasselbe gilt dann natürlich auch für die unmittelbaren Teilaussagen. Eine Folge von Satz 16.4 ist diese schon oben erwähnte Tatsache.

Satz 16.5 *Zu jeder Aussage $\vec{x}$ existiert genau ein p mit $\langle\vec{x}, p\rangle \in U$.*

Beweis. Induktion über die Länge von $\vec{x}$. Der Fall, dass die Länge von $\vec{x}$ Null ist, tritt nicht ein. Ist sie dagegen 1, so ist $\vec{x}$ einfach, und damit ist die Behauptung klar. Sei nun die Länge mindestens 2. Dann existiert genau ein Hauptsymbol. Ist es ¬, so ist $\vec{x} = (\neg\vec{y})$ für ein eindeutiges $\vec{y}$. Damit existiert nach Induktionsvoraussetzung genau ein p mit $\langle\vec{y}, p\rangle \in U$. Damit ist $\langle\vec{x}, -p\rangle \in U$. Es ist daher $\langle\vec{x}, p\rangle \notin U$. Ist das Hauptsymbol ∧, so existieren eindeutig bestimmte $\vec{y}$ und $\vec{z}$ mit $\vec{x} = (\vec{y}\wedge\vec{z})$. Nach Induktionsannahme existieren eindeutig bestimmte p und q mit $\langle\vec{y}, p\rangle, \langle\vec{z}, q\rangle \in U$. Damit ist $\langle(\vec{y}\wedge\vec{z}), p \cap q\rangle \in U$. Und es ist $\langle(\vec{y}\wedge\vec{z}), -(p \cap q)\rangle \notin U$. Ebenso für die anderen Hauptsymbole. ⊣

Daraus folgt, was wir schon angesprochen hatten. Eine gegebene Aussage ist nicht zugleich wahr und falsch.

Nun aber zu dem Beweis des Satzes 16.4. Ich führe dazu die **Klammerbilanz** ein. Ich gebe zunächst eine induktive Definition. Für $s \in A$ sei

$$c(s) := \begin{cases} 1 & \text{falls } s = \texttt{(} \\ -1 & \text{falls } s = \texttt{)} \\ 0 & \text{sonst} \end{cases}$$

Ist $\vec{x} = x_0 x_1 \cdots x_{n-1}$ gegeben, so sei $b_{\vec{x}} : n \to \mathbb{N}$ definiert durch

$$b_{\vec{x}}(i) = c(x_0) + c(x_1) + \cdots + c(x_i)$$

$$(16.6)\qquad \begin{array}{ccccccccccccc} & \texttt{(} & \texttt{(} & \neg & \texttt{w} & \texttt{)} & \wedge & \texttt{(} & \texttt{f} & \vee & \texttt{w} & \texttt{)} & \texttt{)} \\ 0 & 1 & 2 & 2 & 2 & 1 & 1 & 2 & 2 & 2 & 2 & 1 & 0 \end{array}$$

Ist $\vec{x} : n \to A$ eine Zeichenkette, so ist die Bilanz des $i$ten Symbols gerade die Anzahl der öffnenden Klammern minus die Anzahl der schließenden Klammern bis zu diesem Symbol. Eine öffnende Klammer erhöht die Bilanz um eins, eine schließende vermindert sie um eins. Die Bilanz ist immer eine positive Zahl, denn vor jeder schließenden Klammer ist immer eine korrespondierende öffnende.

Lemma 16.6 *Ist ein Zeichenkette $\vec{x}$ eine Aussage, so ist die Bilanz der letzten Position 0, die Bilanz aller Positionen außer der letzten größer als 0.*

Dies gilt im Übrigen auch für einfache Aussagen, denn das einzige Symbol ist auch das letzte, und da ist die Bilanz 0. Dies ist eine starke Eigenschaft von Sprachen. Sie besagt, dass ein Parser von links lesend wissen kann, wann die Zeichenkette zu Ende sein muss. Sie bedeutet auch, dass die Sprache Präfixfrei ist.

Korollar 16.7 *Ist $\vec{x}$ eine Aussage und $\vec{y}$ ein echtes Präfix von $\vec{x}$, so ist $\vec{y}$ keine Aussage.*

Sei nun $\vec{x}$ eine komplexe Zeichenkette.

$$(16.7)\qquad \vec{x} = x_0 x_1 x_2 \cdots x_{n-1}$$

Dann beginnt sie mit einer öffnenden Klammer und endet mit einer schließenden Klammer. Wir unterscheiden zwei Fälle: (a) das zweite Zeichen ist $\neg$. Dann ist $\vec{x} = \texttt{(}\neg\vec{y}\texttt{)}$, wo $\vec{y}$ eindeutig bestimmt ist. (b) Das zweite Zeichen ist von $\neg$ verschieden. Dann hat $\vec{x}$ die Form $\vec{x} = \texttt{(}\vec{u}\texttt{)}$. Es hat $\vec{u}$ die Bilanz 0 (warum?). Ferner ist $\vec{u} = \vec{y}x_j\vec{z}$ für eine gewisses j und Aussagen $\vec{y}$ und $\vec{z}$. Es ist dann $\vec{y}$ die kleinste Zeichenkette $x_1 x_2 \cdots x_{j-1}$, die die Klammerbilanz 0 hat und ist somit eindeutig. Dann ist auch das Operationszeichen, x_j, sowie $\vec{z}$ eindeutig.

Nicht immer ist die eindeutige Lesbarkeit notwendig dafür, dass eine Aussage einen eindeutigen Wahrheitswert hat. Dazu gebe ich ein Beispiel. Wir betrachten Zeichenketten über dem Alphabet $B := \{\texttt{w}, \texttt{f}, \wedge\}$. Aussagen sind jetzt beliebige Folgen der Form $((\texttt{w} \mid \texttt{f})\wedge)^*(\texttt{w} \mid \texttt{f})$, also etwa /w∧f∧f/. Eine Aussage $\vec{x}$ ist genau dann wahr, wenn sie kein Vorkommen von `f` enthält. Dies definiert die Sprache V. Wir können V wiederum induktiv generieren.

Definition 16.8 *Die Menge $T := \{0, 1\}$ ist die Menge der Wahrheitswerte. Es sei V die kleinste Menge, die Folgendes erfüllt.*

① $\langle \texttt{w}, 1\rangle \in V$.

② $\langle \texttt{f}, 0\rangle \in V$.

③ *Falls $\langle \vec{x}, p\rangle \in V$ und $\langle \vec{y}, q\rangle \in V$, so $\langle \vec{x}\texttt{∧}\vec{y}, p \cap q\rangle \in V$.*

Hier gilt nun nicht mehr die eindeutige Lesbarkeit: das Paar $\langle \texttt{w∧f∧w}, 0\rangle$ können wir auf zwei Weisen erzeugen.

1. Aus $\langle \texttt{w}, 1\rangle$ und $\langle \texttt{f}, 0\rangle$ mit der zweiten Regel $\langle \texttt{w∧f}, 0\rangle$ und mit $\langle \texttt{w}, 1\rangle$ schließlich $\langle \texttt{w∧f∧w}, 0\rangle$.

2. Aus $\langle \texttt{f}, 0\rangle$ und $\langle \texttt{w}, 1\rangle$ mit der zweiten Regel $\langle \texttt{f∧w}, 0\rangle$ und mit $\langle \texttt{w}, 1\rangle$ als ersten Teil schließlich $\langle \texttt{w∧f∧w}, 0\rangle$.

Trotzdem gilt weiterhin, dass nicht zugleich $\langle \vec{x}, 1\rangle$ und $\langle \vec{x}, 0\rangle$ in V ist. Das bedeutet, dass man die eindeutige Lesbarkeit nicht braucht, um eindeutige Bedeutung sicherzustellen. Das wird bei Klammerkonventionen wie dieser natürlich benutzt. Ähnlich ist es bei Addition oder Multiplikation.

Im täglichen Umgang mit Formeln wird man bemüht sein, Klammern zu sparen. Das macht sie besser lesbar. Ich will daher folgende Vereinbarung treffen.

1. Die äußeren Klammern dürfen stets wegfallen. Es ist also /`w∧f`/ kurz für /`(w∧f)`/.

2. In einer Negation einer Negation dürfen wir die Klammern weglassen: also wird aus $\texttt{(¬(¬}\vec{x}\texttt{))}$ schlicht $\texttt{(¬¬}\vec{x}\texttt{)}$.

3. In einer Verbindung von einer Konjunktion mit einer Konjunktion dürfen Klammern weggelassen werden. Es gilt Linksklammerung. Das bedeutet, dass der Ausdruck $\vec{x}\texttt{∧}\vec{y}\texttt{∧}\vec{z}$ als $\texttt{((}\vec{x}\texttt{∧}\vec{y}\texttt{)∧}\vec{z}\texttt{)}$ gelesen wird. Es ist also `w∧f∧w` kurz für `((w∧f)∧w)`.

4. Das eben Gesagte trifft auch für die Disjunktion zu.

5. Auch bei der Implikation lassen wir Klammern weg, nur gilt hier Rechtsklammerung. Das bedeutet, dass $\vec{x}\texttt{→}\vec{y}\texttt{→}\vec{z}$ als $\texttt{(}\vec{x}\texttt{→(}\vec{y}\texttt{→}\vec{z}\texttt{))}$ gelesen wird.

6. Negation bindet stärker als Konjunktion, Konjunktion bindet stärker als Disjunktion, Disjunktion bindet stärker als Implikation.

Schließlich vereinbaren wir eine neue Sprache U^+. Es sei $\langle\vec{x}, p\rangle \in U^+$, falls es ein $\langle\vec{y}, p\rangle \in U$ gibt, sodass $\vec{x}$ aus $\vec{y}$ durch Anwendung der oberen Regeln 1 - 6 hervorgeht.

Man mache sich klar, dass unsere Konventionen so gestaltet sind, dass zu jeder Kurzform eine eindeutige Langform gehört. (Und deswegen ist nicht zugleich $\langle\vec{x}, p\rangle \in U$ und $\langle\vec{x}, -p\rangle \in U$.) Zum Beispiel habe ich vereinbart, dass $\vec{x}$→$\vec{y}$→$\vec{z}$ als (`$\vec{x}$→($\vec{y}$→$\vec{z}$)) zu lesen ist; das ist eindeutig, und deswegen muss uns die Tatsache nicht kümmern, dass (($\vec{x}$→$\vec{y}$)→$\vec{z}$) einen anderen Wahrheitswert haben kann; denn wir lesen ja die Formel als rechtsgeklammert, nicht als linksgeklammert. Das Gleiche gilt für Bindungsstärke: `f∧w∨w` ist eindeutig als `((f∧w)∨w)` zu lesen und ist deswegen wahr. Es ist nicht zu lesen als `(f∧(w∨f))` (und wäre somit falsch).

Ich werde zu guter Letzt eine kontextfreie Grammatik angeben, die alle und nur die korrekten, vereinfachten Ausdrücke erzeugt und dabei auch nur die offiziellen Konstituenten zulässt. Mehrfache Klammerung ist nicht gestattet. (Das ließe sich natürlich ändern.)

(16.8)

<Auss> → <BAuss> | <KAuss> | <DAuss> | <PAuss> | <NAuss>
<BAuss> → w | f | (<KAuss>) | (<DAuss>) | (<PAuss>)
| (<NAuss>)
<KAuss> → <BAuss> | <NAuss> | <KAuss>∧<BAuss>
<DAuss> → <BAuss> | <KAuss> | <PAuss> | <NAuss>
| <DAuss>∨<BAuss>
<PAuss> → <BAuss> | <KAuss> | <NAuss> | <BAuss>→<PAuss>
<NAuss> → ¬<BAuss>

Übungen

[56] Bestimmen Sie, ob die folgenden Zeichenketten Aussagen der Sprache U sind. Falls ja, geben Sie ihren Wert an. Falls nein, geben Sie eine kurze Begründung.

1. `(w∧w)`
2. `(w)`
3. `((¬w)∧¬¬w))`
4. `(((¬f)∨(w∧1))∨f))`
5. `((t∨(¬f))∨(¬(¬f)))`

[57] Zu der Zeichenkette `(((f→w)∧((¬f)∧w))→f)` existiert genau ein Wahrheitswert c. Bestimmen Sie ihn und geben Sie eine Ableitung von ⟨`(((f→w)∧((¬f)∧w))→f)`, c⟩.

[58] Geben Sie wie beschrieben die Klammerbilanz an für jedes Symbol in

`((¬f)∧((¬(w∧w))→w))`

Verifizieren Sie die Aussage von Lemma 16.6 für diese Zeichenkette.

[59] Vereinfachen Sie die Formeln ① und ②, indem Sie überflüssige Klammern gemäß den Klammerkonventionen weglassen. Fügen Sie umgekehrt in ③ und ④ Klammern hinzu, bis Sie einen Ausdruck aus U bekommen.

① `((¬w)→((¬w)→f))`

② `(((¬(¬w))∨(f∧f))∧(w∧f))`

③ `¬¬w∨f∧¬w∧¬w→f`

④ `w→f→f→w`

Kapitel 17

Variable und Belegungen

Wir werden unsere Sprache jetzt noch einmal verändern. Jetzt werden wir auch Aussagevariable hinzufügen. Dies wird allerdings auch unsere Semantik verändern, denn Variable haben im Gegensatz zu Konstanten keine vorher festgelegten Werte.

Definition 17.1 (Syntax der Aussagenlogik) *Es sei*

$$A := \{\mathtt{p}, \mathtt{0}, \mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{4}, \mathtt{5}, \mathtt{6}, \mathtt{7}, \mathtt{8}, \mathtt{9}, \mathtt{(}, \mathtt{)}, \mathtt{w}, \mathtt{f}, \neg, \wedge, \vee, \to\} \tag{17.1}$$

unser Alphabet. Die Sprache Auss *ist wie folgt definiert.*

① $\mathtt{p}\vec{x} \in \mathrm{Auss}$*, wo* $\vec{x}$ *eine echte Ziffernfolge ist.*

② $\mathtt{w}, \mathtt{f} \in \mathrm{Auss}$.

③ *Ist* $\vec{x} \in \mathrm{Auss}$*, so ist auch* $(\neg\vec{x}) \in \mathrm{Auss}$.

④ *Sind* $\vec{x}, \vec{y} \in \mathrm{Auss}$*, so auch* $(\vec{x}\wedge\vec{y})$, $(\vec{x}\vee\vec{y})$, $(\vec{x}\to\vec{y}) \in \mathrm{Auss}$.

⑤ *Nichts sonst ist in* Auss.

Eine Zeichenkette aus Auss *wird auch als* ***Aussage*** *bezeichnet.*

Zur Klärung: ich benutze /$\mathtt{0}$/ für die Ziffer „0”, sodass sie auf jeden Fall anders aussieht als der Lateinische Buchstabe „O”, welchen ich mit /O/ wiedergebe.

Die Zeichenketten /$\mathtt{p}\vec{x}$/, wo $\vec{x}$ eine Ziffernfolge ist, heißen auch **Aussagenvariable** (oder schlicht **Variable**, wenn klar ist, dass wir von Aussagen reden). Beispiele sind /$\mathtt{p0}$/, /$\mathtt{p3}$/, /$\mathtt{p17}$/. Wir verabreden, dass die Ziffernfolge $\vec{x}$ diejenige

Zahl repräsentiert, wie sie sonst auch üblich ist. Das bedeutet, dass wir die Variable /`p017`/ nicht als verschieden von der Variablen /`p17`/ ansehen, auch wenn die Zeichenketten verschieden sind. Wer es genau haben möchte: in ① ist damit implizit verabredet, dass $\vec{x}$ nicht mit /`0`/ beginnen darf, es sei denn, $\vec{x} = \mathtt{0}$. (Die Ziffernfolge soll echt sein, das heißt, nicht leer.)

Wiederum können wir diese Definition kompakter aufschreiben, indem wir sie durch eine kontextfreie Grammatik ersetzen.

$$
\begin{array}{rl}
\texttt{<Auss>} \to & \texttt{w} \mid \texttt{f} \mid \texttt{<Var>} \mid \texttt{(}\neg\texttt{<Auss>)} \mid \texttt{(<Auss>}\wedge\texttt{<Auss>)} \\
 & \mid \texttt{(<Auss>}\vee\texttt{<Auss>)} \mid \texttt{(<Auss>}\rightarrow\texttt{<Auss>)} \\
\texttt{<Var>} \to & \texttt{p<Index>} \\
\texttt{<Index>} \to & \texttt{<Ziffer>} \mid \texttt{<EZiffer><Zahl>} \\
\texttt{<Zahl>} \to & \texttt{<Ziffer>} \mid \texttt{<Ziffer><Zahl>} \\
\texttt{<EZiffer>} \to & \texttt{1} \mid \texttt{2} \mid \texttt{3} \mid \texttt{4} \mid \texttt{5} \mid \texttt{6} \mid \texttt{7} \mid \texttt{8} \mid \texttt{9} \\
\texttt{<Ziffer>} \to & \texttt{0} \mid \texttt{1} \mid \texttt{2} \mid \texttt{3} \mid \texttt{4} \mid \texttt{5} \mid \texttt{6} \mid \texttt{7} \mid \texttt{8} \mid \texttt{9}
\end{array}
\tag{17.2}
$$

Die Menge *Auss* ist nunmehr die Menge aller Zeichenketten, die von `<Auss>` abgeleitet werden. Genauso ist die Menge *Var* die Menge aller Zeichenketten, die von `<Var>` abgeleitet werden.

Ich erwähne der Vollständigkeit halber die eindeutige Lesbarkeit.

Lemma 17.2 (Eindeutige Lesbarkeit) *Auss ist eindeutig lesbar. Das bedeutet: ist $\vec{x} \in$ Auss, so tritt genau einer der folgenden Fälle auf.*

① *$\vec{x}$ ist eine Variable.*

② *$\vec{x} = \mathtt{w}$ oder $\mathtt{x} = \mathtt{f}$.*

③ *$\vec{x} = (\neg\vec{y})$ für ein eindeutig bestimmtes $\vec{y}$.*

④ *$\vec{x} = (\vec{y}\wedge\vec{z})$ für eindeutig bestimmte $\vec{y}$ und $\vec{z}$.*

⑤ *$\vec{x} = (\vec{y}\vee\vec{z})$ für eindeutig bestimmte $\vec{y}$ und $\vec{z}$.*

⑥ *$\vec{x} = (\vec{y}\rightarrow\vec{z})$ für eindeutig bestimmte $\vec{y}$ und $\vec{z}$.*

Der Beweis ist eine leichte Modifikation des Arguments aus dem vorigen Kapitel.

Definition 17.3 (Semantik der Aussagenlogik) *Eine Belegung ist eine Funktion $\beta : \mathrm{Var} \to \{0, 1\}$, wo* Var *die Menge der Variablen ist. Eine Aussage hat einen Wert $\overline{\beta}(\vec{x})$ unter der Belegung β, der wie folgt errechnet wird.*

① $\overline{\beta}(\vec{x}) := \beta(\vec{x})$*, falls* $\vec{x}$ *eine Variable ist.*

② $\overline{\beta}(\mathtt{f}) = 0.$

③ $\overline{\beta}(\mathtt{w}) = 1.$

④ $\overline{\beta}(\mathtt{(\neg}\vec{x}\mathtt{)}) := -\overline{\beta}(\vec{x}).$

⑤ $\overline{\beta}(\mathtt{(}\vec{x}\mathtt{\wedge}\vec{y}\mathtt{)}) := \overline{\beta}(\vec{x}) \cap \overline{\beta}(\vec{y}).$

⑥ $\overline{\beta}(\mathtt{(}\vec{x}\mathtt{\vee}\vec{y}\mathtt{)}) := \overline{\beta}(\vec{x}) \cup \overline{\beta}(\vec{y}).$

⑦ $\overline{\beta}(\mathtt{(}\vec{x}\mathtt{\rightarrow}\vec{y}\mathtt{)}) := \overline{\beta}(\vec{x}) \rightarrow \overline{\beta}(\vec{y}).$

β ***erfüllt*** $\vec{x}$*, falls* $\overline{\beta}(\vec{x}) = 1$*. Es bezeichnet* Bel *die Menge aller Belegungen. Schließlich führe ich folgende Schreibweise ein.*

(17.3) $$[\vec{x}] := \{\beta \in Bel : \overline{\beta}(\vec{x}) = 1\}$$

Man beachte, dass $\overline{\beta} : Auss \to \{0, 1\}$. Es ist also $\overline{\beta}$ eine Funktion, die jeder Aussage einen Wahrheitswert zuordnet. β hingegen gibt lediglich jeder Variablen einen Wert. Man schreibt allerdings des Öfteren einfach $\beta(\vec{x})$ anstelle von $\overline{\beta}(\vec{x})$, weil die Unterscheidung zwischen diesen beiden Funktionen oft nicht wesentlich ist. Ist $\overline{\beta} = \overline{\gamma}$, so muss natürlich $\beta = \gamma$ sein. Das Umgekehrte ist auch richtig, wie wir jetzt zeigen.

Proposition 17.4 *Wenn* $\beta = \gamma$*, so ist* $\overline{\beta} = \overline{\gamma}$*.*

Beweis. Wir zeigen $\overline{\beta}(\vec{x}) = \overline{\gamma}(\vec{x})$ für alle $\vec{x} \in Auss$. Der Beweis ist induktiv und verwendet die eindeutige Lesbarkeit. (Das ist auch einer der Gründe, warum man eindeutige Lesbarkeit so dringend braucht.) Nehmen wir zunächst an, $\vec{x}$ sei nicht zusammengesetzt. Sei $\vec{x}$ eine Variable. Dann ist nach Voraussetzung $\overline{\beta}(\vec{x}) = \beta(\vec{x}) = \gamma(\vec{x}) = \overline{\gamma}(\vec{x})$. Oder $\vec{x} = \mathtt{w}$ oder $\vec{x} = \mathtt{f}$. In diesem Fall ist dann $\overline{\beta}(\vec{x}) = \overline{\gamma}(\vec{x})$ wegen ② bzw. ③. Sei nun $\vec{x}$ zusammengesetzt. Dann ist $\vec{x} = \mathtt{(\neg}\vec{y}\mathtt{)}$, oder $\vec{x} = \mathtt{(}\vec{y}\mathtt{\vee}\vec{z}\mathtt{)}$, oder $\vec{x} = \mathtt{(}\vec{y}\mathtt{\wedge}\vec{z}\mathtt{)}$, oder $\vec{x} = \mathtt{(}\vec{y}\mathtt{\rightarrow}\vec{z}\mathtt{)}$. Stets tritt genau einer der Fälle auf, und $\vec{y}$ bzw. $\vec{z}$ sind dann ebenfalls eindeutig bestimmt. Deswegen ist nach Induktionsvoraussetzung $\overline{\beta}(\vec{y}) = \overline{\gamma}(\vec{y})$ bzw. $\overline{\beta}(\vec{z}) = \overline{\gamma}(\vec{z})$ und somit $\overline{\beta}(\vec{x}) = \overline{\gamma}(\vec{x})$. ⊣

Zunächst noch eine wichtige Definition.

Definition 17.5 *Eine Aussage* $\vec{x}$ *heißt* ***Tautologie****, falls sie unter jeder Belegung erfüllt wird.* $\vec{x}$ *heißt* ***Kontradiktion****, wenn sie unter keiner Belegung erfüllt wird. Ist* $\vec{x}$ *weder Tautologie noch Kontradiktion, so heißt* $\vec{x}$ ***kontingent****.*

Erst ein paar Beispiele. Sei β : $\mathtt{p0} \mapsto 0, \mathtt{p1} \mapsto 0, \mathtt{p2} \mapsto 1, \mathtt{p3} \mapsto 1$ und sei $\beta(p) := 0$ sonst. Dies ist also eine Belegung. Ebenso ist γ : $\mathtt{p0} \mapsto 1, \mathtt{p1} \mapsto 0, \mathtt{p2} \mapsto 1, \mathtt{p3} \mapsto 1$ (und $\gamma(p) := 0$ sonst) ebenfalls eine Belegung, die sich aber nur in dem Wert für p0 von β unterscheidet.

1. $\overline{\beta}(\mathtt{(p0\to p1)}) = \overline{\beta}(\mathtt{p0}) \to \overline{\beta}(\mathtt{p1}) = 0 \to 0 = 1$.

2. $\overline{\beta}(\mathtt{(p0\vee(\neg p2))}) = \overline{\beta}(\mathtt{p0}) \cup \overline{\beta}(\mathtt{(\neg p2)}) = 0 \cup (-\overline{\beta}(\mathtt{p2})) = 0 \cup (-1) = 0 \cup 0 = 0$.

3. $\overline{\gamma}(\mathtt{(p0\to p1)}) = \overline{\gamma}(\mathtt{p0}) \to \overline{\gamma}(\mathtt{p1}) = 1 \to 0 = 0$.

4. $\overline{\gamma}(\mathtt{(p0\vee(\neg p2))}) = \overline{\gamma}(\mathtt{p0}) \cup \overline{\gamma}(\mathtt{(\neg p2)}) = 1 \cup (-\overline{\gamma}(\mathtt{p2})) = 1 \cup (-1) = 1 \cup 0 = 1$.

Dazu ein paar Bemerkungen. Technisch gesehen ist β eine Funktion von Zeichenketten nach Wahrheitswerten. Diese Zeichenketten sind unsere Variable. In der Literatur werden diese oft wie folgt geschrieben: p_0, p_1, p_2, usf. Dabei wird die Natur der Indizes nicht weiter problematisiert. Man nimmt einfach an, dass sie Zahlen sind. Zahlen werden aber im täglichen Leben durch Ziffernfolgen repräsentiert. Dass diese Folgen nicht die Zahlen *sind*, muss man sich erst einmal klarmachen. Dazu genügt es, sich in der Welt all die verschiedenen Zahlsysteme anzuschauen; all diese bezeichnen stets die Zahlen, tun dies aber auf verschiedene Weise. In Kapitel 15 hatte ich zum Beispiel das Dualsystem vorgestellt. Dort ist /LOLOL/ ein Name für die Zahl 21. Hier benutze ich nun die Folge /21/, um ebendiese Zahl darzustellen. Insofern merkt man gar keinen richtigen Unterschied. Das ist jetzt volle Absicht: ich will diesen Unterschied gar nicht hochspielen; wer also hier so tut, als sei die Ziffernfolge schon die Zahl selbst, macht keinen Fehler. In der Tat halte ich es hier genau so. Aber ich will der Genauigkeit halber auf den Punkt aufmerksam gemacht haben. Wir sagen also hier einfach, Variable sind irgendwelche speziellen Zeichenketten. Aussagen sind auch Zeichenketten. Allerdings ist nicht jede Aussage eine Variable.

Zweitens habe ich die Definition des Werts unter einer Belegung in die gängige Form gegossen. So findet man das in den Lehrbüchern. Allerdings darf man nur dann so definieren, wenn klar ist, dass diese Definition eindeutig ist. Es gilt aber nach wie vor eindeutige Lesbarkeit. Die eindeutige Lesbarkeit garantiert dann, wie oben gezeigt, dass $\overline{\beta}(\vec{x})$ nur einen einzigen Wert hat. Ich weise auf ein wichtiges Faktum hin. Es besagt, dass der Wert einer Formel unter einer Belegung nur von Werten derjenigen Variablen abhängt, die in der Formel auftreten. Das muss ich noch definieren. Eine Variable kommt in einer Formel vor, wenn die Formel mit Hilfe (unter anderem) dieser Variable aufgebaut wurde; dazu muss die Variable zumindest einmal eine Teilkette sein, etwa /`p0`/ in /`(p0→p3)`/. Aber das genügt noch nicht. Wir betrachten dazu die Formel /`((¬p17)→p22)`/. Intuitiv sieht man leicht ein, dass diese Formel aus den Variablen /`p17`/ und /`p22`/ aufgebaut ist, nicht jedoch aus den Variablen /`p1`/ bzw. /`p2`/. Diese letzteren sind Teilketten; jedoch ist ihr Vorkommen derart, dass immer noch eine Ziffer folgt.

Definition 17.6 (Vorkommen) *Es sei $\vec{x}$ eine Formel und $\vec{u}$ eine Variable. $\vec{u}$* ***kommt als Variable in*** $\vec{x}$ ***vor****, wenn $\vec{x}$ eine Zerlegung $\vec{x} = \vec{y}\vec{u}\vec{z}$ besitzt, sodass $\vec{z}$ nicht mit einer Ziffer beginnt. Eine Formel $\vec{u}$ kommt in $\vec{x}$ vor, wenn sie entweder eine Variable ist und als Variable vorkommt oder eine Formel ist (aber keine Variable) und als Zeichenkette vorkommt.*

Der Unterschied zwischen Vorkommen als Zeichenkette und Vorkommen als Variable ist wesentlich. Außerdem müssen wir stets und immer von Vorkommen reden. Jedes Vorkommen einer Variable (als Variable) ist auch ein Vorkommen dieser Variable als Zeichenkette, aber umgekehrt gilt *nicht*, dass jedes Vorkommen dieser Variable als Zeichenkette auch eines als Variable ist. Man beachte zum Beispiel, dass in /`((¬p17)→p1)`/ die Variable /`p1`/ vorkommt; aber sie kommt nur *einmal* vor. Das erste Vorkommen der Kette /`p1`/ ist *kein* Vorkommen dieser Variable *als Variable*. Nur das zweite Vorkommen der Kette ist auch ein Vorkommen der Variable. Man beachte auch, dass ich von einem Vorkommen einer Variable rede, wenn ich meine, es sei ein Vorkommen einer Variable $\vec{x}$ *als Variable*; ich rede dahingegen von einem Vorkommen von $\vec{x}$ als Zeichenkette, wenn es als Zeichenkette vorkommt. Schließlich und endlich mache ich darauf aufmerksam, dass für Zeichenketten, die keine Variable sind, keinerlei Vorsicht nötig ist. Wenn man eine solche Zeichenkette findet, kommt sie dort bereits als Formel vor und nicht nur als Zeichenkette.

Das Ganze gießen wir in eine Definition.

Definition 17.7 (Vorkommende Variablen) *Die Menge* $\mathrm{Var}(\vec{x})$ *der in $\vec{x}$ auftretenden Variablen, ist wie folgt definiert.*

② $\mathrm{Var}(\mathtt{w}) = \mathrm{Var}(\mathtt{f}) = \varnothing$;

③ $\mathrm{Var}(\vec{x}) = \{\vec{x}\}$, *falls* $\vec{x} \in \mathrm{Var}$;

④ $\mathrm{Var}(\mathtt{(}\neg\vec{x}\mathtt{)}) = \mathrm{Var}(\vec{x})$;

⑤ $\mathrm{Var}(\mathtt{(}\vec{x}\wedge\vec{y}\mathtt{)}) = \mathrm{Var}(\vec{x}) \cup \mathrm{Var}(\vec{y})$;

⑥ $\mathrm{Var}(\texttt{(}\vec{x}\texttt{∨}\vec{y}\texttt{)}) = \mathrm{Var}(\vec{x}) \cup \mathrm{Var}(\vec{y})$;

⑦ $\mathrm{Var}(\texttt{(}\vec{x}\texttt{→}\vec{y}\texttt{)}) = \mathrm{Var}(\vec{x}) \cup \mathrm{Var}(\vec{y})$.

Lemma 17.8 (Koinzidenzlemma) *Es sei $\vec{x}$ eine Formel und β und γ zwei Belegungen mit $\beta(p) = \gamma(p)$ für alle $p \in \mathrm{Var}(\vec{x})$, dh für alle in $\vec{x}$ vorkommenden Variablen. Dann ist $\overline{\beta}(\vec{x}) = \overline{\gamma}(\vec{x})$.*

Beweis. Man muss dieses Lemma wie alle Behauptungen beweisen. Dazu gehe ich die Fälle der Reihe nach durch. Der Beweis erfolgt durch Induktion über den Formelaufbau. Sei $\vec{x}$ eine Variable. Dann kommt allein $\vec{x}$ in $\vec{x}$ (als Variable) vor. Nach Annahme ist deswegen schon $\overline{\beta}(\vec{x}) = \beta(\vec{x}) = \gamma(\vec{x}) = \overline{\gamma}(\vec{x})$. Sei nun $\vec{x} = \texttt{w}$ oder $\vec{x} = \texttt{f}$. Dann enthält $\vec{x}$ keine Variable, und nach Definition ist in beiden Fällen $\overline{\beta}(\vec{x}) = \overline{\gamma}(\vec{x})$. Sei jetzt $\vec{x}$ zusammengesetzt. Hat $\vec{x}$ die Form $\texttt{(¬}\vec{y}\texttt{)}$, so kommt eine Variable in $\vec{x}$ genau dann vor, wenn sie in $\vec{y}$ vorkommt. Nach Induktionsannahme ist $\overline{\beta}(\vec{y}) = \overline{\gamma}(\vec{y})$. Nun ist $\overline{\beta}(\vec{x}) = 1-\overline{\beta}(\vec{y}) = 1-\overline{\gamma}(\vec{y}) = \overline{\gamma}(\vec{x})$. Hat $\vec{x}$ die Form $\texttt{(}\vec{y}\texttt{∧}\vec{z}\texttt{)}$, so kommt eine Variable in $\vec{x}$ genau dann vor, wenn sie $\vec{y}$ oder in $\vec{z}$ vorkommt. Nach Induktionsvoraussetzung ist deswegen $\overline{\beta}(\vec{y}) = \overline{\gamma}(\vec{y})$ (da ja β und γ auf allen in $\vec{y}$ vorkommenden Variablen übereinstimmen) und aus dem analogen Grund $\overline{\beta}(\vec{z}) = \overline{\gamma}(\vec{z})$. Deswegen haben wir jetzt $\overline{\beta}(\vec{x}) = \overline{\beta}(\vec{y}) \cap \overline{\beta}(\vec{z}) = \overline{\gamma}(\vec{y}) \cap \overline{\gamma}(\vec{z}) = \overline{\gamma}(\vec{x})$. Die anderen Fälle sind völlig analog. ⊣

Satz 17.9 *Sei Bel die Menge der Belegungen. Es gilt:*

① $[p\vec{x}] = \{\beta : \beta(p\vec{x}) = 1\}$.

② $[\texttt{f}] = \varnothing$, $[\texttt{w}] = Bel$.

③ $[\texttt{(¬}\vec{x}\texttt{)}] = Bel - [\vec{x}]$.

④ $[\texttt{(}\vec{x}\texttt{∧}\vec{y}\texttt{)}] = [\vec{x}] \cap [\vec{y}]$.

⑤ $[\texttt{(}\vec{x}\texttt{∨}\vec{y}\texttt{)}] = [\vec{x}] \cup [\vec{y}]$.

⑥ $[\texttt{(}\vec{x}\texttt{→}\vec{y}\texttt{)}] = (Bel - [\vec{x}]) \cup [\vec{y}]$.

Der Beweis ist einfach. Ich führe das exemplarisch vor. $\beta \in [\texttt{(¬}\vec{x}\texttt{)}]$ genau dann, wenn $\overline{\beta}(\texttt{(¬}\vec{x}\texttt{)}) = 1$ genau dann, wenn $\overline{\beta}(\vec{x}) = 0$. Letzteres ist genau dann der Fall, wenn $\beta \notin [\vec{x}]$, dh $\beta \in Bel - [\vec{x}]$.

Definition 17.10 *Wir schreiben „$\vec{x} \approx \vec{y}$”, falls $[\vec{x}] = [\vec{y}]$. Wir sagen in diesem Fall, die Aussagen seien (**semantisch**) **gleichwertig** oder **synonym**.*

Es ist also genau dann „$\vec{x} \approx \vec{y}$" wahr, wenn für jede Belegung β gilt: $\beta(\vec{x}) = \beta(\vec{y})$. So ist $\mathtt{f}$ eine Kontradiktion und $\mathtt{w}$ eine Tautologie. Es ist klar, dass $\vec{x}$ genau dann eine Tautologie (Kontradiktion) ist, wenn $\vec{x} \approx \mathtt{w}$ ($\vec{x} \approx \mathtt{f}$).

Die Gleichheit lässt sich durch ausprobieren leicht feststellen. Das ist die Methode der Wahrheitstafeln. Auf der anderen Seite können wir Gleichungen auch rein formal ableiten. Dazu dient ein Kalkül. Bevor ich ihn vorstelle, gebe ich erst einmal ein paar Gleichungen an, die allgemein, das heißt für alle Aussagen $\vec{x}$, $\vec{y}$ und $\vec{z}$ gelten.

$$\begin{array}{llll}
(\vec{x}\wedge(\vec{y}\wedge\vec{z})) & \approx ((\vec{x}\wedge\vec{y})\wedge\vec{z}) & (\vec{x}\vee(\vec{y}\vee\vec{z})) & \approx ((\vec{x}\vee\vec{y})\vee\vec{z}) \\
(\vec{x}\wedge\vec{y}) & \approx (\vec{y}\wedge\vec{x}) & (\vec{x}\vee\vec{y}) & \approx (\vec{y}\vee\vec{x}) \\
(\vec{x}\wedge\vec{x}) & \approx \vec{x} & (\vec{x}\vee\vec{x}) & \approx \vec{x} \\
(\vec{x}\wedge\mathtt{f}) & \approx \mathtt{f} & (\vec{x}\wedge\mathtt{w}) & \approx \vec{x} \\
(\vec{x}\vee\mathtt{f}) & \approx \vec{x} & (\vec{x}\vee\mathtt{w}) & \approx \mathtt{w} \\
((\vec{x}\wedge\vec{y})\vee\vec{x}) & \approx \vec{x} & ((\vec{x}\vee\vec{y})\wedge\vec{x}) & \approx \vec{x} \\
(\vec{x}\wedge(\vec{y}\vee\vec{z})) & \approx ((\vec{x}\wedge\vec{y})\vee(\vec{x}\wedge\vec{z})) & (\vec{x}\vee(\vec{y}\wedge\vec{z})) & \approx ((\vec{x}\vee\vec{y})\wedge(\vec{x}\vee\vec{z})) \\
(\vec{x}\wedge(\neg\vec{x})) & \approx \mathtt{f} & (\vec{x}\vee(\neg\vec{x})) & \approx \mathtt{w} \\
(\neg(\neg\vec{x})) & \approx \vec{x} & (\neg\mathtt{w}) & \approx \mathtt{f} \\
(\neg\mathtt{f}) & \approx \mathtt{w} & (\vec{x}\to\vec{y}) & \approx ((\neg\vec{x})\vee\vec{y}) \\
(\neg(\vec{x}\wedge\vec{y})) & \approx ((\neg\vec{x})\vee(\neg\vec{y})) & (\neg(\vec{x}\vee\vec{y})) & \approx ((\neg\vec{x})\wedge(\neg\vec{y}))
\end{array} \tag{17.4}$$

Die erste Zeile besagt, dass $\wedge$ und $\vee$ **assoziativ** sind, die zweite Zeile, dass $\wedge$ und $\vee$ **kommutativ** sind. Die Gesetze der dritten Zeile sind die **Absorptionsgesetze**, die der fünften sind die **Distributivgesetze**. Die Gesetze der letzten Zeile heißen auch **de Morgansche Gesetze**.

Diese Gesetze kann man natürlich beweisen. Ich führe dies an dem ersten de Morganschen Gesetz vor. Es ist $(\neg(\vec{x}\wedge\vec{y})) \approx ((\neg\vec{x})\vee(\neg\vec{y}))$ definiert durch $[(\neg(\vec{x}\wedge\vec{y}))] = [((\neg\vec{x})\vee(\neg\vec{y}))]$, was dann der Fall ist, wenn für alle Belegungen β: $\overline{\beta}((\neg(\vec{x}\wedge\vec{y}))) = \overline{\beta}(((\neg\vec{x})\vee(\neg\vec{y})))$. Dies kann man fallweise durchgehen.

1. $\overline{\beta}(\vec{x}) = 1$ und $\overline{\beta}(\vec{y}) = 1$. Dann ist $\overline{\beta}((\neg(\vec{x}\wedge\vec{y}))) = 0$ sowie $\overline{\beta}(((\neg\vec{x})\vee(\neg\vec{y}))) = 0$.

2. $\overline{\beta}(\vec{x}) = 1$ und $\overline{\beta}(\vec{y}) = 0$. Dann ist $\overline{\beta}((\neg(\vec{x}\wedge\vec{y}))) = 1$ sowie $\overline{\beta}(((\neg\vec{x})\vee(\neg\vec{y}))) = 1$.

3. $\overline{\beta}(\vec{x}) = 0$ und $\overline{\beta}(\vec{y}) = 1$. Dann ist $\overline{\beta}((\neg(\vec{x}\wedge\vec{y}))) = 1$ sowie $\overline{\beta}(((\neg\vec{x})\vee(\neg\vec{y}))) = 1$.

4. $\overline{\beta}(\vec{x}) = 0$ und $\overline{\beta}(\vec{y}) = 0$. Dann ist $\overline{\beta}((\neg(\vec{x}\wedge\vec{y}))) = 1$ sowie $\overline{\beta}(((\neg\vec{x})\vee(\neg\vec{y}))) = 1$.

Mit den Gleichungen lässt sich nun wie mit Gleichungen für Zahlausdrücke rechnen. In die Gleichungen darf man für die Variable (also $\vec{x}$, $\vec{y}$ und $\vec{z}$) beliebige Aussagen einsetzen und bekommt gültige Gleichungen über Aussagen. Hier ist ein informelles Beispiel.

$$\begin{aligned}\texttt{(p0}\wedge\texttt{((}\neg\texttt{p1)}\vee\texttt{(}\neg\texttt{p0)))} &\approx \texttt{((p0}\wedge\texttt{(}\neg\texttt{p1))}\vee\texttt{(p0}\wedge\texttt{(}\neg\texttt{p0)))}\\ &\approx \texttt{((p0}\wedge\texttt{(}\neg\texttt{p1))}\vee\texttt{f)}\\ &\approx \texttt{(p0}\wedge\texttt{(}\neg\texttt{p1))}\end{aligned} \tag{17.5}$$

Daraus folgt jetzt zum Beispiel die Gültigkeit folgender Gleichung.

$$\texttt{(p0}\wedge\texttt{((}\neg\texttt{p1)}\vee\texttt{(}\neg\texttt{p0)))} \approx \texttt{(p0}\wedge\texttt{(}\neg\texttt{p1))} \tag{17.6}$$

Die Gleichungen sind durchaus nicht unabhängig voneinander. So kann man $\texttt{(}\vec{x}\vee\texttt{(}\neg\vec{x}\texttt{))} \approx \texttt{w}$ aus den anderen Gleichungen ableiten. Das geht wie folgt.

$$\begin{aligned}\texttt{(p0}\vee\texttt{(}\neg\texttt{p0))} &\approx \texttt{(}\neg\texttt{(}\neg\texttt{(p0}\vee\texttt{(}\neg\texttt{p0))))}\\ &\approx \texttt{(}\neg\texttt{((}\neg\texttt{p0)}\wedge\texttt{(}\neg\texttt{(}\neg\texttt{p0))))}\\ &\approx \texttt{(}\neg\texttt{f)}\\ &\approx \texttt{w}\end{aligned} \tag{17.7}$$

Dazu wurde zunächst die doppelte Negation angewendet, anschließend ein de Morgansches Gesetz. Im dritten Schritt wurde die Konjunktion einer Formel und ihres Negats durch /$\texttt{f}$/ ersetzt. Und als letztes haben wir benutzt, dass das Negat von /$\texttt{f}$/ gerade /$\texttt{w}$/ ist.

Dies kann man auch schematisch aufschreiben, dh ohne den Gebrauch konkreter Aussagen:

$$\begin{aligned}\texttt{(}\vec{x}\vee\texttt{(}\neg\vec{x}\texttt{))} &\approx \texttt{(}\neg\texttt{(}\neg\texttt{(}\vec{x}\wedge\texttt{(}\neg\vec{x}\texttt{))))}\\ &\approx \texttt{(}\neg\texttt{(((}\neg\vec{x}\texttt{)}\wedge\texttt{((}\neg\texttt{(}\neg\vec{x}\texttt{))))}\\ &\approx \texttt{(}\neg\texttt{f)}\\ &\approx \texttt{w}\end{aligned} \tag{17.8}$$

Formal beschreibt man das so. Es gelten ganz allgemein folgende Regeln.

$$\vdash \vec{x} \approx \vec{x} \qquad \frac{\vdash \vec{x} \approx \vec{y}}{\vdash \vec{y} \approx \vec{x}} \qquad \frac{\vdash \vec{x} \approx \vec{y} \qquad \vdash \vec{y} \approx \vec{z}}{\vdash \vec{x} \approx \vec{z}} \tag{17.9}$$

Hierbei ist $\vdash$ das sogenannte **Urteilszeichen**. Es besagt, dass die Gleichung **ableitbar** ist, nicht, dass sie wahr ist. Dass eine Gleichung wahr ist, schreiben wir ohne $\vdash$.

Ferner gilt noch folgende Regel. Es sei dazu $\vec{z}$ eine Zeichenkette, die eine Variable, sagen wir p enthält. Dann bezeichnet $[\vec{x}/p]\vec{z}$ das Ergebnis der Ersetzung aller Vorkommen von p als Variable in $\vec{z}$ durch $\vec{x}$. Zum Beispiel ist

(17.10) `[((¬p2)→f)/p0](p0→((¬p1)∨p0))`

nichts weiter als die Formel

(17.11) `(((¬p2)→f)→((¬p1)∨((¬p2)→f)))`

Ersetzung bedeutet in diesem Zusammenhang Ersetzung von Vorkommen einer Zeichenkette als Variable durch eine andere. Das heißt: überall, wo die Zeichenkette „`p0`” zu sehen ist und als Variable vorkommt, wird sie ausgeschnitten und durch „`((¬p2)→f)`” ersetzt. Man mache sich klar, dass die ersetzende Zeichenkette ebenfalls als Formel vorkommt. Denn ist sie Variable, so folgt ihr keine Ziffer, und ist sie Formel aber keine Variable, so kommt sie bereits als Formel vor.

$$\frac{\vdash \vec{x} \approx \vec{y}}{\vdash [\vec{x}/p]\vec{z} \approx [\vec{y}/p]\vec{z}} \tag{17.12}$$

Definition 17.11 (Gleichungskalkül) *Es ist „$\vdash \vec{x} \approx \vec{y}$” wahr, falls entweder $\vec{x} \approx \vec{y}$ eine Substitutionsinstanz einer Gleichung in der Liste* (17.4) *ist, oder mit Hilfe einer der Regeln von* (17.9) *und* (17.12) *aus einer bereits ableitbaren Gleichung hervorgeht.*

Diese Definition lässt sich noch weiter präzisieren, aber ich werde das nicht tun. Formales Beweisen werden wir später ohnehin noch behandeln. Es ist sicher klar, dass diese Regeln korrekt sind: wenn wir $\vdash \vec{x} \approx \vec{y}$ ableiten können, so ist auch $\vec{x} \approx \vec{y}$. Dass die Umkehrung gilt, ist nicht so leicht zu sehen. Ich verschiebe das auf das nächste Kapitel, verkünde aber schon das Ergebnis.

Satz 17.12 (Adäquatheit des Gleichungskalküls) *Genau dann ist $\vdash \vec{x} \approx \vec{y}$ aus* (17.4) *mit Hilfe von* (17.9) *und* (17.12) *ableitbar, wenn $\vec{x} \approx \vec{y}$.*

Übungen

[60] Es sei $\beta(\mathrm{p}\vec{x}) = 1$ genau dann, wenn die duch $\vec{x}$ dargestellte Zahl 0 ist oder ungerade. Berechnen Sie

1. $\overline{\beta}(\mathtt{p1{\wedge}\neg p0})$
2. $\overline{\beta}(\mathtt{(p0{\rightarrow}p1){\rightarrow}\neg p0})$
3. $\overline{\beta}(\mathtt{(p0{\rightarrow}p1){\wedge}(\neg p3{\rightarrow}p0)})$
4. $\overline{\beta}(\mathtt{p1{\rightarrow}\neg p1})$

Hinweis. Sie müssen die Klammerkonventionen beachten.

[61] Es sei γ eine Belegung mit $\gamma(\mathrm{p}\vec{x}) = 1$ genau dann, wenn $\vec{x}$ durch 3 teilbar ist. Ferner sei $\beta(\mathrm{p}\vec{x}) = 1$ genau dann, wenn $\vec{x}$ kleiner als 7 ist. Können Sie ohne die Wahrheitswerte auszurechnen sagen, ob $\overline{\beta}(\mathtt{p7{\rightarrow}(p0{\vee}\neg p6)}) = \overline{\gamma}(\mathtt{p7{\rightarrow}(p0{\vee}\neg p6)})$? (Vergessen Sie nicht, Ihre Antwort zu begründen!)

[62] Es sei $\vec{x}$ eine Aussage und β eine Belegung derart, dass $\overline{\beta}(\neg\vec{x}{\rightarrow}\vec{x}) = 1$. Zeigen Sie, dass $\overline{\beta}(\vec{x}) = 1$. (Dies ist die sogenannte *Regel des Clavius*.)

[63] Zeigen Sie, dass $\mathtt{((p1{\rightarrow}p2){\rightarrow}p1){\rightarrow}p1} \approx \mathtt{w}$. *Anleitung.* Sie können entweder eine Herleitung aus den Gesetzen versuchen oder die Belegungen durchprobieren.

Kapitel 18

Normalformen

Von nun an wende ich die Klammerkonventionen an und schreibe auch β anstelle von $\overline{\beta}$. Das bedeutet, dass Klammern nur dort gesetzt werden, wo sie wirklich nötig sind. Dies bedeutet in diesem Kapitel eine massive Ersparnis an Klammern. Damit augenfällig wird, dass wir von den wirklichen Zeichenketten abstrahieren, benutze ich, wie in anderen Büchern üblich, kleine griechische Buchstaben (vorzugsweise φ, ψ, χ und δ), um Formeln zu symbolisieren.

Zunächst will ich eine Notation einführen. Seien φ_i, $i < n$, irgendwelche Formeln. Dann setze:

$$\bigwedge_{i<n} \varphi_i := \varphi_0 \wedge \varphi_1 \wedge \cdots \wedge \varphi_{n-1} \tag{18.1}$$

Wir nennen das die Konjunktion über die φ_i, $i < n$, auch wenn es Sonderfälle gibt, in denen womöglich gar kein Konjunktionsszeichen erscheint. Denn man muss die Fälle $n = 0$ und $n = 1$ gesondert betrachten. Im Fall $n = 0$ setzen wir $\bigwedge_{i<0} \varphi_i := \mathtt{w}$. Im Fall $n = 1$ setzen wir $\bigwedge_{i<1} \varphi_i := \varphi_0$ (kein Konjunktionszeichen).

Dies lässt sich auch rekursiv definieren.

$$\begin{aligned} \textstyle\bigwedge_{i<0} \varphi_i &:= \mathtt{w} \\ \textstyle\bigwedge_{i<1} \varphi_i &:= \varphi_0 \\ \textstyle\bigwedge_{i<n+1} \varphi_i &:= (\textstyle\bigwedge_{i<n} \varphi_i) \wedge \varphi_n \end{aligned} \tag{18.2}$$

(Die zweite Zeile ist nicht überflüssig, weil die Rekursionsklausel für $n = 0$ nicht das richtige Ergebnis liefern würde. Wir würden dann nämlich $\mathtt{w} \wedge \varphi_0$ bekommen, was nicht unserer Intention entspricht. Man beachte auch, dass die klammerfreie Konjunktion als linksgeklammert aufgefasst wird. Die Rekursion ist entsprechend aufgebaut.)

Ist H eine Menge von Formeln, so schreiben wir auch

$$\bigwedge_{\varphi \in H} \varphi \tag{18.3}$$

und meinen damit die Konjunktion aller Formeln aus H, wobei irgendeine Reihenfolge gewählt werden darf. Konkret sieht das so aus. Wir nehmen an, $H = \{\chi_0, \chi_1, \cdots, \chi_{n-1}\}$, ohne Doppelnennungen. Und dann ist

$$\bigwedge_{\varphi \in H} \varphi = \chi_0 \wedge \chi_1 \wedge \cdots \wedge \chi_{n-1} \tag{18.4}$$

Ebenso setzen wir

$$\begin{array}{ll} \bigvee_{i<0} \varphi_i & := \mathtt{f} \\ \bigvee_{i<1} \varphi_i & := \varphi_0 \\ \bigvee_{i<n+1} \varphi_i & := (\bigvee_{i<n} \varphi_i) \vee \varphi_n \end{array} \tag{18.5}$$

(Und wiederum ist die zweite Zeile nötig, weil in der dritten Zeile $n > 0$ sein muss, ansonsten bekommen wir ein überflüssiges Vorkommen von `w`.)

Kommen wir nun zu dem Problem der Normalform.

Definition 18.1 (Literal) *Eine Aussage φ ist ein **Literal**, wenn sie eine Variable oder eine negierte Variable ist.*

Es sei φ eine Konjunktion von Literalen, etwa

$$\varphi = \mathtt{p0} \wedge \mathtt{p2} \wedge \neg \mathtt{p1} \wedge \neg \mathtt{p4} \tag{18.6}$$

In φ dürfen Klammern komplett wegfallen. Denn die Negation bindet stärker als Konjunktion, weswegen `¬p1∧¬p4` eindeutig als `((¬p1)∧(¬p4))` gelesen werden muss.

Wir können leicht ablesen, welche Belegung β solch eine Konjunktion φ erfüllt. Wir müssen dabei nur auf die auftretenden Literale schauen. In unserem Beispiel muss gelten:

$$\beta(\mathtt{p0}) = 1, \beta(\mathtt{p2}) = 1, \beta(\mathtt{p1}) = 0, \beta(\mathtt{p4}) = 0 \tag{18.7}$$

Ist χ eine Variable, so muss $\beta(\chi) = 1$ sein, wenn sie in φ lediglich positiv, das heißt also positiv and nicht negativ, auftritt; tritt sie lediglich negativ auf, so muss $\beta(\chi) = 0$ sein. Falls sie gar nicht auftritt, so darf $\beta(\chi) = 0$ oder $\beta(\chi) = 1$ sein. Denn dann hängt der Wert von φ unter β nicht von $\beta(\chi)$ ab (das sagt das Koinzidenzlemma). Tritt schließlich die Variable sowohl negativ wie positiv auf, so ist die Formel φ nicht erfüllbar.

Die Formel (18.6) ist unter anderem synonym mit

(18.8) `p2∧¬p1∧p0∧¬p4`

Desweiteren kann eine Variable sowohl positiv als auch negativ auftreten, etwa in

(18.9) `p0∧p2∧¬p0¬p1∧¬p4`

In diesem Fall ist die Formel nicht erfüllbar.

Definition 18.2 (Minimale Konjunktion) *Eine **minimale Konjunktion** ist eine Konjunktion von Literalen, bei der jede Variable höchstens einmal auftritt.*

Ich merke an, dass, wenn eine Variable nur einmal auftritt, sie insbesondere nur positiv oder nur negativ auftritt.

Zusätzlich können wir die Variable nach ihrer Ordnungsnummer aufsteigend ordnen. Dann würde aus (18.6) die folgende Formel.

(18.10) `p0∧¬p1∧p2∧¬p4`

Definition 18.3 (Disjunktive Normalform) *Eine Formel ist in **disjunktiver Normalform** (**DNF**), falls sie die Formel `f` ist, eine minimale Konjunktion oder eine Disjunktion von paarweise verschiedenen minimalen Konjunktionen ist.*

Etwas knapper formuliert: eine Formel φ ist in disjunktiver Normalform, falls es eine Menge $H = \{\chi_i : i < n\}$ von minimalen Konjunktionen χ_i gibt derart, dass $\varphi = \bigvee_{i<n} \chi_i$.

Satz 18.4 *Zu jeder Formel φ existiert eine Formel χ in disjunktiver Normalform mit $\varphi \approx \chi$.*

Beweis. Erster Fall: φ ist variablenfrei. Dann ist φ entweder synonym mit `f` oder synonym mit `w`. Das erste ist eine leere Disjunktion, das zweite ist eine leere Konjunktion, also ebenfalls in disjunktiver Normalform (setze $H := \{\texttt{w}\}$).

Nun sei die Menge U der Variablen von φ nicht leer. Zu jeder Teilmenge $P \subseteq U$ sei nun

$$\lambda_P := \bigwedge_{p \in P} p \wedge \bigwedge_{p \in U-P} \neg p \tag{18.11}$$

Hierbei existieren zwei Sonderfälle. Ist $P = \varnothing$, so setze anstelle von (18.11)

(18.12) $$\lambda_P := \bigwedge_{p \in U-P} \neg p$$

und ist $P = U$, so setze

(18.13) $$\lambda_P := \bigwedge_{p \in P} p$$

Damit ist λ_P stets eine minimale Konjunktion.

Nun sei β_P eine Belegung, für die $\beta_P(p) = 1$, falls $p \in P$ und $\beta_P(p) = 0$, falls $p \in U - P$. Dann gilt $\beta_P(\lambda_P) = 1$, und es ist $\beta_P(\lambda_Q) = 0$, falls $Q \neq P$. Endlich sei

(18.14) $$\chi := \bigvee_{\beta_P(\varphi)=1} \lambda_P$$

Dann ist $\beta_P(\chi) = 1$ genau dann, wenn ein Q existiert mit $\beta_Q(\varphi) = 1$ und $\beta_P(\lambda_Q) = 1$. Dies ist genau dann der Fall, wenn ein Q existiert mit $\beta_Q(\varphi) = 1$ und $Q = P$, genau dann, wenn $\beta_P(\varphi) = 1$. Und das war zu zeigen. ⊣

Dies lässt sich sogar noch schärfer formulieren.

Satz 18.5 *Zu jeder nichkontradiktorischen Formel φ existiert eine Formel χ in disjunktiver Normalform in denselben Variablen wie φ mit $\varphi \approx \chi$.*

Ist die Formel φ eine Kontradiktion, so ist die DNF zwangsläufig `f` (die Disjunktion über die leere Menge), und hat deswegen keine Variable.

Ich gebe ein Beispiel. Es sei die folgende Formel gegeben.

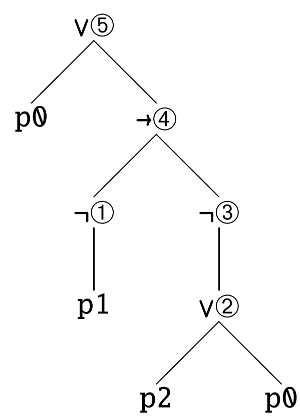
(18.15) `(p0∨(¬p1→¬(p2∨p0)))`

Wir listen auf, unter welchen Belegungen die Formel erfüllt ist. Dazu bedienen wir uns der Methode der **Wahrheitstafeln**, die ich im Folgenden erkläre. Wir beginnen damit, die Variablen links aufzulisten und jede mögliche Belegung hinzuschreiben. Anschließend schreiben wir unter jedes Vorkommen der Variablen den Wert unter der Belegung:

(18.16)

p0	p1	p2	(p0	∨	(¬	p1	→	¬	(p2	∨	p0)))
0	0	0	0			0			0		0
0	0	1	0			0			1		0
0	1	0	0			1			0		0
0	1	1	0			1			1		0
1	0	0	1			0			0		1
1	0	1	1			0			1		1
1	1	0	1			1			0		1
1	1	1	1			1			1		1

Wir zählen die Operationszeichen von innen nach außen ab. Das bedeutet, dass die am weitesten eingebetteten Vorkommen die kleinsten Nummern haben. Das Hauptsymbol hat dagegen die höchste. Diese Aufzählung sagt uns, in welcher Reihenfolge wir die Werte der Teilformeln ausrechnen müssen. Dies ergibt die Ziffern ①, ②, ③ und ④ in der obersten Zeile der untenstehenden Tabelle. (Diese Ziffern sind nicht nötig, aber nützlich, damit man nicht durcheinanderkommt.) Am Besten sieht man dies am Syntaxbaum:



In jedem Knoten ist die angezeigte Zahl größer als in allen von ihm dominierten Knoten. Das bedeutet, dass der Wahrheitswert der zugehörigen Konstituente dieses Knotens später ausgerechnet wird als der seiner Teilkonstituenten. (Man überlege sich, dass es auch andere Nummerierungen gibt, die diese Eigenschaft haben.)

Für jede Teilformel rechnen wir den Wahrheitswert aus und schreiben ihn unter das Hauptsymbol der Formel:

(18.17)

				⑤	①		④	③		②	
`p0`	`p1`	`p2`	`(p0`	`∨`	`(¬`	`p1`	`→`	`¬`	`(p2`	`∨`	`p0)))`
0	0	0	0	1	1	0	1	1	0	0	0
0	0	1	0	0	1	0	0	0	1	1	0
0	1	0	0	1	0	1	1	1	0	0	0
0	1	1	0	1	0	1	1	0	1	1	0
1	0	0	1	1	1	0	0	0	0	1	1
1	0	1	1	1	1	0	0	0	1	1	1
1	1	0	1	1	0	1	1	0	0	1	1
1	1	1	1	1	0	1	1	0	1	1	1

Wir sehen also in der ersten und neunten Spalte (die Zählung beginnt nach dem senkrechten Strich!) den Wert der Teilformel `p0`, in der vierten Spalte den Wert von `p1`, und in der siebten den Wert von `p2`. In der dritten Spalte vermerken wir den Wert von `(¬p1)`, den wir als erstes berechnen (daher die ① darüber). In der achten Spalte notieren wir den Wert von `(p2∨p0)`, den wir als zweites berechnen (②). In der sechsten Spalte schließlich notieren wir den Wert von `(¬(p2∨p0))` (③), anschließend den Pfeil (④) und, zu guter Letzt, in der zweiten Spalte den Wert der gesamten Formel (⑤). Ausschlaggebend ist also die zweite Spalte (Hauptsymbol: `∨`). Dort sehen wir den Wert der Formel unter der Belegung.

Der Aufbau einer Wahrheitstafel ist intuitiv recht einfach. Wir listen die Werte der Variablen links auf und füllen die Tabelle schrittweise. Die Auflistung der Werte sollte streng nach einem festgelegten Schema erfolgen, weil man sonst eine Zeile vergisst. In der $n$ten Zeile steht der Binärcode der Zahl $n-1$, wobei die Anzahl Ziffern genau die der Anzahl der Variablen ist. Wir beginnen also in der 1. Zeile mit dem Code von 0, und der ist `000` (führende Nullen werden hier hingeschrieben, damit die Länge genau drei ist). Dann folgt in der 2. Zeile der Code von 1, also `001`, und so weiter. Bei k Variablen sind es genau 2^k Zeilen.

Mit Hilfe der Wahrheitstafeln können wir auch sehen, ob zwei Formeln synonym sind, dh ob $\varphi \approx \chi$. Dazu listen wir die Werte unter allen Variablen, die in φ oder χ auftreten und rechnen dann den Wert von φ und χ aus. Finden wir in jeder Zeile denselben Wert, so gilt $\varphi \approx \chi$. Schauen wir uns zum Beispiel die Formel `(p0∨(p2→p1))` an.

(18.18)

p0	p1	p2	(p0	∨	(p2	→	p1))
0	0	0	0	1	0	1	0
0	0	1	0	0	1	0	0
0	1	0	0	1	0	1	1
0	1	1	0	1	1	1	1
1	0	0	1	1	0	1	0
1	0	1	1	1	1	0	0
1	1	0	1	1	0	1	1
1	1	1	1	1	1	1	1

Vergleichen wir das Ergebnis (Spalte 2) mit dem obigen (ebenfalls in Spalte 2), so sehen wir stets die gleichen Werte. Also sind die Aussagen synonym.

Schauen wir uns noch einmal die Wahrheitstafel an, und zwar wie gehabt die zweite Spalte, die unter dem Hauptsymbol steht. Wenn dort eine 1 steht, wird die noch zu konstruierende Formel λ_P als Disjunkt in die DNF aufgenommen. Die erste Zeile gehört dazu. Ich erinnere noch einmal an die Definition (18.11). P ist die Menge der Variablen unserer Formel, die wir erfüllen *müssen*. Diejenigen Variable, die nicht in P sind aber in der Formel auftreten, dürfen wir nicht erfüllen. In der ersten Zeile ist $P = \varnothing$, da die Belegung keiner Variablen den Wert 1 zuordnet. Es ist nun λ_P die Konjunktion aller Variablen, die in P auftreten und der *Negation* aller anderen Variablen aus U, dh derjenigen, die in der Formel aber nicht in P auftreten. Die Formel λ_P lautet somit für $P = \varnothing$

(18.19) $\lambda_\varnothing = \neg\mathtt{p0}\wedge\neg\mathtt{p1}\wedge\neg\mathtt{p2}$

Für die dritte Zeile ist $P = \{\mathtt{p1}\}$. Damit ist

(18.20) $\lambda_{\{\mathtt{p1}\}} = \neg\mathtt{p0}\wedge\mathtt{p1}\wedge\neg\mathtt{p2}$

Und so weiter. Einzig die zweite Zeile (also $P = \{\mathtt{p2}\}$) wird ausgelassen. Damit hat unsere DNF die Form

(18.21) $\lambda_\varnothing\vee\lambda_{\{\mathtt{p1}\}}\vee\lambda_{\{\mathtt{p1},\mathtt{p2}\}}\vee\lambda_{\{\mathtt{p0}\}}\vee\lambda_{\{\mathtt{p0},\mathtt{p2}\}}\vee\lambda_{\{\mathtt{p0},\mathtt{p1}\}}\vee\lambda_{\{\mathtt{p0},\mathtt{p1},\mathtt{p2}\}}$

Hier nun ist die DNF:

(18.22)
$$\begin{array}{llll} & \neg\mathtt{p0}\wedge\neg\mathtt{p1}\wedge\neg\mathtt{p2} & \vee & \neg\mathtt{p0}\wedge\mathtt{p1}\wedge\neg\mathtt{p2} \\ \vee & \neg\mathtt{p0}\wedge\mathtt{p1}\wedge\mathtt{p2} & \vee & \mathtt{p0}\wedge\neg\mathtt{p1}\wedge\neg\mathtt{p2} \\ \vee & \mathtt{p0}\wedge\neg\mathtt{p1}\wedge\mathtt{p2} & \vee & \mathtt{p0}\wedge\mathtt{p1}\wedge\neg\mathtt{p2} \\ \vee & \mathtt{p0}\wedge\mathtt{p1}\wedge\mathtt{p2} & & \end{array}$$

Ich gebe an dieser Stelle noch Folgendes zu bedenken. Ich spreche zwar von *der* DNF, aber diese ist nicht wirklich eindeutig. Die Differenzen sind aber nicht der Rede wert.

Die erstellte DNF hat die Eigenschaft, dass sämtliche Disjunktionsglieder alle und nur die Variablen enthalten, die in der Formel auftreten. Dies ist die maximal explizite Form. Wenn das so ist, muss ich noch erlauben, dass `w` seine eigene DNF ist (warum?).

Man beachte, dass wir die DNF (18.22) sehr weit vereinfachen können. Es ist ja $\varphi\wedge\chi\vee\varphi\wedge\neg\chi$ semantisch gleichwertig mit φ. Wir können die obige Form also nach dieser Gleichung in mehreren Schritten auf die Form (18.23) zusammenziehen.

(18.23)
```
     ¬p0∧¬p2
  ∨  ¬p0∧p1∧p2
  ∨  p0
```

Dies lässt sich leichter lesen und gibt uns ebenfalls eine leicht lesbare Bedingung an die Belegung.

Es gibt außer der Disjunktiven Normalform auch noch eine Konjunktive Normalform (KNF). Sie unterscheidet sich von der DNF dadurch, dass Disjunktion und Konjunktion vertauscht sind.

Definition 18.6 (Minimale Disjunktion) *Eine* ***minimale Disjunktion*** *ist eine Disjunktion von Literalen, bei der jede Variable höchstens einmal auftritt.*

Definition 18.7 (Konjunktive Normalform) *Eine Formel ist in* ***konjunktiver Normalform (KNF)****, falls sie* `w` *ist oder eine Konjunktion von paarweise verschiedenen minimalen Disjunktionen.*

Satz 18.8 *Zu jeder Formel φ existiert eine Formel χ in konjunktiver Normalform mit $\varphi \approx \chi$. Ist φ keine Tautologie, so kann χ sogar so gewählt werden, dass* $\mathrm{Var}(\varphi) = \mathrm{Var}(\chi)$.

Beweis. Ähnlich. Es seien V die in φ auftretenden Variablen. Sei β eine Belegung, die φ nicht erfüllt. Dann setze

$$(18.24) \qquad \delta_\beta := \bigvee_{p\in V,\beta(p)=0} p \vee \bigvee_{p\in V,\beta(p)=1} \neg p$$

Dann gilt $\gamma(\delta_\beta) = 0$ genau dann, wenn $\gamma(p) = \beta(p)$ für alle $p \in V$. Damit wird

$$(18.25) \qquad \chi := \bigwedge_{\beta(\varphi)=0} \delta_\beta$$

Dies ist die gesuchte Formel. Ist φ eine Tautologie, so ist die Menge $\{\beta : \beta(\varphi) = 0\}$ leer, also steht rechts `w` nach Definition. Ansonsten steht eine Konjunktion von Disjunktionen von Literalen. ⊣

Dies ergibt folgendes Rezept für die Erstellung einer KNF. Wir nehmen die Disjunktion χ aller Konjunktionen für Belegungen, die φ *nicht* erfüllen. Dies ist dann exakt die DNF für $\neg\varphi$. Dann ist $\neg\chi$ synonym mit φ. Man ziehe die Negation nach innen: die Negation der Disjunktionen wird eine Konjunktion von negierten Konjunktionen. Die letzteren sind Disjunktionen von negierten und doppelt negierten Variablen. Streiche schließlich die doppelte Negation, und die KNF ist fertig. Ich gebe ein Beispiel. Es sei die folgende Formel gegeben:

(18.26) `¬(p1∨p2)∨¬(p1→¬p2)`

Wir erstellen zunächst unsere Tabelle.

(18.27)

p1	p2	¬	(p1	∨	p2)	∨	¬	(p1	→	¬	p2)
0	0	1	0	0	0	1	1	0	1	1	0
0	1	0	0	1	1	0	0	0	1	0	1
1	0	0	1	1	0	0	0	1	1	1	0
1	1	0	1	1	0	1	0	1	0	0	1

Wir sammeln die nicht erfüllenden Belegungen und schreiben so eine DNF für die Negation:

(18.28) `¬p1∧p2∨p1∧¬p2`

Jetzt formen die Negation dieser Formel um:

$$(18.29) \qquad \begin{aligned} \texttt{¬(¬p1∧p2∨p1∧¬p2)} &= \texttt{(¬(¬p1∧p2))∧¬(p1∧¬p2)} \\ &= \texttt{(¬¬p1∨¬p2)∧(¬p1∨¬¬p2)} \\ &= \texttt{(p1∨¬p2)∧(¬p1∨p2)} \end{aligned}$$

Dies ist, wie gesagt, die KNF zu der *ursprünglichen* Formel, (18.26)!

Ich kehre nun zum Beweis der Vollständigkeit des Gleichungskalküls zurück. Diese besagt, dass jeder Sachverhalt $\vec{x} \approx \vec{y}$ aus den gegeben Gleichungen ableitbar ist.

Der Beweis geht wie folt. Im ersten Schritt zeigt man, dass für jedes $\vec{x}$ und jede disjunktive Normalform $\vec{y}$ von $\vec{x}$ die Behauptung „$\vdash \vec{x} \approx \vec{y}$" ableitbar ist. Der Rest ist dann einfach: genau dann ist $\vec{x} \approx \vec{z}$, wenn es eine DNF $\vec{y}$ gibt, die sowohl von $\vec{x}$ wie auch von $\vec{z}$ eine DNF ist. Da jetzt folgt, dass sowohl $\vdash \vec{x} \approx \vec{y}$ als auch $\vdash \vec{z} \approx \vec{y}$ ableitbar sind, so ist jetzt $\vdash \vec{y} \approx \vec{z}$ ableitbar und damit $\vdash \vec{x} \approx \vec{z}$. Wie kommt man also zu der gewünschten DNF? Das ist recht langwierig, aber relativ einfach. Wir können $\rightarrow$ eliminieren, weil wir die Gleichung $(\vec{x}\rightarrow\vec{y}) \approx ((\neg\vec{x})\vee\vec{y})$ haben. Mittels der de Morganschen Gesetze können wir Negation ganz nach innen schieben, und dann doppelte Negation löschen. Anschließend verwenden wir die Distributivgesetze, um Vorkommen von Disjunktion innerhalb von Konjunktionen zu eliminieren. Jetzt sind wir fast fertig. Wir haben eine Disjunktion von Konjunktionen von Literalen. Falls eine Konjunktion eine Variable sowohl positiv als auch negativ enthält, dürfen wir die Konjunktion durch `f` ersetzen, und in der Disjunktion eliminieren (sofern es nicht das einzige Disjkuntionsglied ist). Mehrfache Vorkommen dürfen auch eliminiert werden. All das ist durch die Gleichungen (17.4) abgedeckt.

Übungen

[64] Zeigen Sie mit Hilfe der Methode der Wahrheitstafeln, dass

$$(\neg(p1\wedge\neg p7)\vee p3)\rightarrow(\neg p3\rightarrow(p1\rightarrow p7))$$

und `w` synonym sind. Geben Sie für die Formel eine DNF an.

[65] Bestimmen Sie alle Belegungen, die die folgende Formel erfüllen und bestimmen Sie für die Formel eine DNF:

$$(p2\rightarrow p0)\rightarrow(p1\rightarrow(p0\vee\neg p1)).$$

[66] Geben Sie zwei DNF φ und χ an, sodass $\varphi\vee\chi$ *keine* DNF ist.

[67] Geben Sie eine DNF und eine KNF an zu $((p1\rightarrow p2)\rightarrow(\neg p0))\rightarrow p0$.

Kapitel 19

Der Tableau-Kalkül

Der **Tableau-Kalkül** ist eine Variation der Methode der Wahrheitstafeln. Im Wesentlichen sind Tableaus erfunden worden, weil die Methode der Wahrheitstafeln in der Regel sehr umständlich ist. Betrachten wir ein einfaches Beispiel, die Formel

(19.1) $\quad$ `p1∧¬p0∧(p1→p0)`

Benutzen wir Wahrheitstafeln, müssen wir 4 Zeilen schreiben, und 8 Spalten mit Wahrheitswerten füllen. Dabei sieht man leicht, dass die Formel nicht erfüllbar ist. Tableaus sind eine Methode, möglichst schnell einen Widerspruch zu entdecken, wenn er vorliegt. Wenn nicht, dann wollen wenigstens *eine* erfüllende Belegung erhalten. Dazu fragen wir uns: ist die gegebene Formel erfüllbar und wenn ja, wie? Eine Konjunktion aber ist nur erfüllbar, wenn jedes seiner Konjunktionsglieder erfüllbar ist, und eine Belegung, die die Konjunktion erfüllt, erfüllt jedes der Konjunktionsglieder. Wir suchen also nach Belegungen, die die folgende Menge von Formeln erfüllt in dem Sinne, dass jede von ihnen wahr wird.

(19.2) $\quad$ $M := \{\texttt{p1}, \neg\texttt{p0}, \texttt{p1}\to\texttt{p0}\}$

Wir wissen in diesem Moment schon, dass die Belegung `p1` wahr machen muss und `p0` falsch. In diesem Fall können wir sofort berechnen, dass `p1→p0` falsch ist und somit keine Belegung existiert, die die Menge erfüllt. Aber wir können das auch etwas systematischer ausrechnen. Wir überlegen so: eine Belegung, die `p1→p0` erfüllt, macht `p1` falsch oder aber `p0` wahr. Das bedeutet, dass eine Belegung, die M erfüllt, eine der beiden Mengen N_1, N_2 erfüllen muss.

(19.3) $\quad$ $N_1 := \{\texttt{p1}, \neg\texttt{p0}, \neg\texttt{p1}\}, \qquad N_2 := \{\texttt{p1}, \neg\texttt{p0}, \texttt{p0}\}$

Es ist unmittelbar klar, dass keine der beiden Mengen erfüllbar sind. Denn sowohl N_1 wie auch N_2 enthalten eine Variable und ihre Negation. Ebenfalls sollte klar sein, dass eine Belegung genau dann M erfüllt, wenn sie entweder N_1 oder N_2 erfüllt. Denn N_1 und N_2 unterscheiden sich von M nur dadurch, dass sie die Formel `p1→p0` nicht enthalten aber dafür einmal `¬p1` und das andere Mal `p0`. Und wie wir wissen, kann man `p1→p0` dadurch erfüllen, dass man `p1` nicht erfüllt oder aber `p0` erfüllt.

Bevor ich nun den vollständigen Tableau-Kalkül vorstelle, will ich noch die Notation vorstellen. Wie in der Logik allgemein üblich, verzichtet man auf Mengenklammern und benutzt das Semikolon als Trennzeichen. Die Menge M wird also wie folgt notiert:

(19.4) `p1; ¬p0; p1→p0`

Sind zwei Mengen alternativ zu betrachten, dann werden sie durch einen senkrechten Strich getrennt:

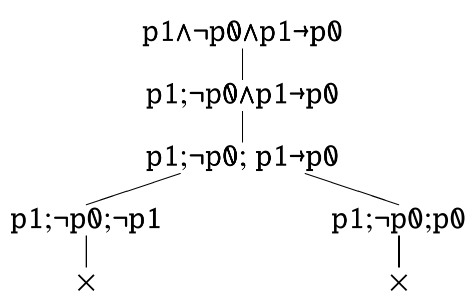
(19.5) `p1; ¬p0; ¬p1 | p1; ¬p0; p0`

Wir sind nun bereit, unser erstes Tableau aufzuschreiben:

(19.6)
$$\begin{array}{c}
\mathtt{p1\land\neg p0\land p1\to p0} \\ \hline
\mathtt{p1;\ \neg p0\land p1\to p0} \\ \hline
\mathtt{p1;\ \neg p0;\ p1\to p0} \\ \hline
\mathtt{p1;\ \neg p0;\ \neg p1 | p1;\ \neg p0;\ p0} \\ \hline
\times \quad | \quad \times
\end{array}$$

Die Kreuze am Ende der Ableitung bedeuten, dass die Mengen nicht erfüllbar sind. Wir sagen, dass das Tableau **schließt**, wenn alle Spalten mit einem Kreuz enden. In diesem Fall ist die Ausgangsmenge nicht erfüllbar. Das angegebene Tableau schließt.

Eine alternative Schreibweise ist in Form eines verzweigenden Baumes wie folgt.



Hierbei rechnen wir von oben nach unten. Sobald es mehrere Möglichkeiten gibt, fügen wir eine Verzweigung ein.

Ein Tableau besteht also darin, eine Formelmenge so abzubauen, dass sie am Ende nur noch Variable und ihre Negationen enthält. Dann können wir unmittelbar sagen, ob sie erfüllbar ist oder nicht. Nun gelingt es nicht immer, aus der Menge eine einzelne Menge dieser Art zu erzeugen. Manchmal müssen wir mehrere Fälle unterscheiden. In diesem Fall kommen wir zu Mengenalternativen. Das bedeutet, dass in dem Tableau jede Zeile aus einer Reihe von Mengen besteht. Dabei verstehen wir jede der Mengen als eine Alternative. Wir suchen also nach Belegungen, die *eine oder mehrere* der Alternativen erfüllt, wobei eine Belegung eine Menge erfüllt, wenn sie jede in ihr auftretende Formel erfüllt.

Ich beginne nun mit der Aufzählung der Tableau-Regeln. Zunächst die Regeln für Konjunktion.

$$\text{(19.7)} \qquad \frac{\Delta;\varphi\wedge\chi}{\Delta;\varphi;\chi} \qquad \frac{\Delta;\neg(\varphi\wedge\chi)}{\Delta;\neg\varphi \mid \Delta;\neg\chi}$$

Dies bedeutet Folgendes: anstelle einer Menge, die aus irgendwelchen Formeln (die wir in Δ sammeln) mitsamt der Formel $\varphi\wedge\chi$ besteht, betrachten wir die Menge Δ zusammen mit den Formeln φ und χ. Δ darf natürlich leer sein. Die zweite Regel sagt, anstelle der Menge Δ mitsamt der Formel $\neg(\varphi\wedge\chi)$ betrachten wir die Alternativen $\Delta;\neg\varphi$ und $\Delta;\neg\chi$. Diese Alternativen können wir auch als Zweige in einem Baum darstellen. Das sieht dann so aus:

$$\begin{array}{ccc} & \Delta;\neg(\varphi\wedge\chi) & \\ \swarrow & & \searrow \\ \Delta;\neg\varphi & & \Delta;\neg\chi \end{array}$$

Dies bedeutet, dass wir, sobald wir eine Menge der Form $\Delta;\neg(\varphi\wedge\chi)$ haben, eine Verzweigung einrichten und in die beiden Zweige die Mengen $\Delta;\neg\varphi$ und $\Delta;\neg\chi$ eintragen. In jedem Zweig rechnen wir dann getrennt weiter.

Wie für $\wedge$ gibt es zwei Regeln zum Abbau von $\vee$.

$$\text{(19.8)} \qquad \frac{\Delta;\neg(\varphi\vee\chi)}{\Delta;\neg\varphi;\neg\chi} \qquad \frac{\Delta;\varphi\vee\chi}{\Delta;\varphi \mid \Delta;\chi}$$

Und es gibt zwei Regeln zum Abbau von $\rightarrow$.

$$\text{(19.9)} \qquad \frac{\Delta;\neg(\varphi\rightarrow\chi)}{\Delta;\varphi;\neg\chi} \qquad \frac{\Delta;\varphi\rightarrow\chi}{\Delta;\neg\varphi \mid \Delta;\chi}$$

Zu guter Letzt brauchen wir Regeln, die die Negation abbauen. Hier gibt es nur eine, nämlich

$$\text{(19.10)} \qquad \frac{\Delta; \neg\neg\varphi}{\Delta; \varphi}$$

Damit lassen sich alle Formeln soweit abbauen, dass wir nur Variable oder deren Negation zu stehen haben. Denn enthält die Menge eine andere Formel, so ist das Hauptsymbol entweder ∧, ∨, → oder ¬. In den ersten drei Fällen haben wir jeweils eine Abbauregel angegeben. Im letzten Fall hat die Formel die Form $\neg\varphi$. Ist φ keine Variable, so hat sie ein Hauptsymbol. Ist dies ∨, ∧ oder →, so haben wir wiederum eine Abbauregel angegeben. Ist es ¬, so hat die Formel nunmehr die Form $\neg\neg\varphi$, und genau dafür haben wir auch eine Abbauregel.

Was ist also zu tun, wenn wir eine Menge aus Variablen und ihren Negationen vor uns haben? Hier gibt es zwei Fälle:

① Die Menge enthält eine Variable und ihre Negation. Dann ist sie nicht erfüllbar. In diesem Fall schreiben wir darunter ein Kreuz.

② Die Menge enthält jede Variable entweder nur positiv oder nur negativ. Eine Belegung ist dann wie folgt konstruiert: $\beta(p) := 1$, wenn die Menge die Variable p enthält, $\beta(p) := 0$, wenn die Menge die Variable negiert enthält. Enthält die Menge die Variable weder positiv noch negativ, so können wir den Wert beliebig festsetzen.

Wir können aber auch anders verfahren und die folgende Regel hinschreiben.

$$\text{(19.11)} \qquad \frac{\Delta; \varphi; \neg\varphi}{\times}$$

Falls wir auch noch die Symbole `w` und `f` haben, gelten noch folgende Regeln.

$$\text{(19.12)} \qquad \frac{\Delta; \mathtt{w}}{\Delta} \qquad \frac{\Delta; \neg\mathtt{w}}{\times} \qquad \frac{\Delta; \mathtt{f}}{\times} \qquad \frac{\Delta; \neg\mathtt{f}}{\Delta}$$

Wenden wir unseren Kalkül auf die Formel `¬(p0→(p1→p0))` an.

$$\text{(19.13)} \qquad \begin{array}{c} \mathtt{\neg(p0\to(p1\to p0))} \\ \hline \mathtt{p0}; \mathtt{\neg(p1\to p0)} \\ \hline \mathtt{p0}; \mathtt{p1}; \mathtt{\neg p0} \\ \hline \times \end{array}$$

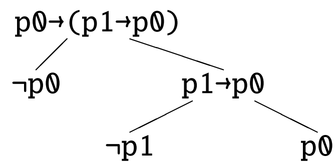
Das Tableau beginnt mit einer Formel, welche allerdings genauer die Menge ist, die diese Formel als einziges Element enthält. Das Tableau endet damit, dass seine einzige Alternative schließt. Es gibt also keine Belegung, die diese Formel erfüllt.

Schauen wir uns die unnegierte Formel an.

(19.14)

p0→(p1→p0)		
¬p0	p1→p0	
¬p0	¬p1	p0

In Baumform:



Wir bekommen im Ergebnis, dass es drei Alternativen gibt: eine Belegung, die ¬p0 erfüllt, eine Belegung, die p0 erfüllt und eine dritte, die ¬p1 erfüllt. Damit ist alles gesagt. Man kann sich in diesem Fall weiter überlegen, dass eine Belegung ja immer p0 erfüllt oder nicht erfüllt, insofern die drei Bedingungen zusammen, da sie ja alternativ gedacht sind, gar keine echte Bedingung sind: jede Belegung erfüllt eine der Alternativen mithin unsere Formel. Sie ist ein Axiom.

Ich fasse zusammen:

> Eine Zeile im Tableau besteht aus einer oder mehreren Mengen. Eine Belegung erfüllt eine Zeile, wenn sie eine der in ihr enthaltenen Mengen erfüllt. Eine Belegung erfüllt eine Menge, wenn sie jede in ihr auftretende Formel erfüllt. Ein Schritt in einem Tableau besteht darin, auf eine in einer Zeile befindliche Menge eine der oben genannten Regeln anzuwenden. Der Kalkül endet, wenn keine der Regeln anwendbar ist. Das Tableau schließt, wenn jede Alternative schließt (d.h. nur aus $\times$ besteht). Andernfalls schließt das Tableau nicht, ist also offen. In diesem Fall gibt es mindestens eine Belegung, die die Ausgangszeile erfüllt. Genauer erfüllt eine Belegung die Anfangszeile genau dann, wenn sie die Endzeile erfüllt.

Der Witz an dem Tableau-Kalkül ist, dass jeder Übergang eine Äquivalenzumformung ist. Ein Belegung, die die Alternativen in einer Zeile erfüllt, erfüllt auch die Alternativen in der nächsten Zeile und umgekehrt.

Es kann nun im Verlauf der Ableitung vorkommen, dass wir die Wahl haben, welche Formel wir abbauen. Hier gibt es keine Vorschriften, womit wir anfangen sollen. Alle Wahlen sind gleich gut. Allerdings ist es ratsam, immer solche Regeln zu bevorzugen, die keine Alternativen eröffnen. Denn damit erspart man sich Schreibarbeit.

Betrachten wir zum Vergleich eine Formel aus dem vorigen Kapitel.

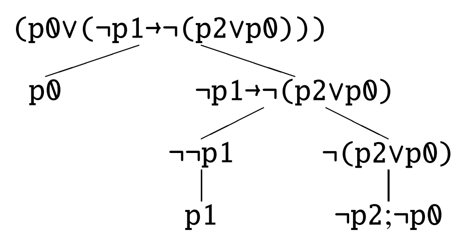
(19.15) `(p0∨(¬p1→¬(p2∨p0)))`

Hier ist ein Tableau:

(19.16)

(p0∨(¬p1→¬(p2∨p0)))		
p0	¬p1→¬(p2∨p0)))	
p0	¬¬p1	¬(p2∨p0)))
p0	p1	¬(p2∨p0)))
p0	p1	¬p2; ¬p0

Wiederum gebe ich die alternative Form als Baum an.



Man sieht hier recht deutlich, warum die Schreibweise als Baum bevorzugt wird. Sie ist kompakter, weil man die Alternativen nicht ständig kopieren muss. Beide Schreibweisen sind natürlich vollständig gleichwertig.

Ich komme noch einmal auf die Wahrheitstafeln zurück. Die Wahrheitstafeln waren ja ein Hilfsmittel, wie wir die erfüllenden Belegungen alle aufzählen konnten. Als Ergebnis bekamen wir eine DNF. Der Tableau-Kalkül funktioniert ähnlich. Zunächst einmal ist klar, dass, wenn das Tableau fertig ist, die letzte Zeile fast unmittelbar eine DNF liefert. Betrachten wir einen Zweig, der nicht schließt. Dann befinden sich dort am Ende eine Menge Δ von Literalen (sind es keine Literale, ließe sich noch eine Regel anwenden). In dieser Menge tritt eine Variable nur einmal auf: entweder positiv oder negativ. Eine Belegung erfüllt Δ, wenn sie jedes Literal erfüllt. Die zugehörige Konjunktion ist nichts anderes als $\bigwedge \Delta$! Man beachte, dass $\bigwedge \Delta = \mathtt{w}$, wenn Δ leer ist (was vorkommen kann, siehe die Regeln (19.12)). Die gesuchte DNF ist die Disjunktion über alle $\bigwedge \Delta$, wo Δ weder leer noch geschlossen ist.

Ist die Formel eine Kontradiktion, gibt es keinen offenen Zweig. Das sieht man unmittelbar. In diesem Fall ist die DNF gerade gleich $\mathtt{f}$. In den anderen Fällen ist die DNF, die man bekommt, allerdings anders als im Fall der Wahrheitstafeln. Sie ist reduziert. Ich gebe noch ein Beispiel, die Formel von Peirce.

(19.17)
$$\begin{array}{c|c|c} \multicolumn{3}{c}{\mathtt{((p0\to p1)\to p0)\to p0}} \\ \hline \multicolumn{2}{c|}{\mathtt{\neg((p0\to p1)\to p0)}} & \mathtt{p0} \\ \hline \multicolumn{2}{c|}{\mathtt{p0\to p1;\neg p0}} & \mathtt{p0} \\ \hline \mathtt{\neg p0;\neg p0} & \mathtt{p1;\neg p0} & \mathtt{p0} \end{array}$$

Zunächst einmal ist die Menge am ersten Zweig nichts anderes als $\mathtt{\neg p0}$. Es handelt sich um Mengen, deswegen ist erneutes Hinschreiben derselben Formel unnötig. Wir haben also die DNF

(19.18) $\mathtt{\neg p0\vee p1\wedge\neg p0\vee p0}$

Man kann das zweite Disjunktionsglied aber weglassen. Ist $\Delta \subseteq \Delta'$, so kann man $\bigwedge \Delta'$ stets weglassen, sofern man $\bigwedge \Delta$ behält. Also bekommen wir

(19.19) $\mathtt{\neg p0\vee p0}$

Dies wiederum reduziert sich zu $\mathtt{w}$. Selbstverständlich gibt es auch kompliziertere Formeln, aber ich verzichte auf eine eingehende Diskussion. Bei den Wahrheitstafeln ist die Auflistung der Fälle so explizit, dass wir sehen können, ob die Formel eine Tautologie ist. Hier muss man einfach eine geduldige Fallunterscheidung vornehmen. Ähnlich sieht es mit der Frage aus, ob wir mit Hilfe von Tableaus entscheiden können, ob zwei Formeln φ und χ synonym sind. Auch hier ist die Wahrheitstafel am Ende eindeutig. Man kann bei der Interpretation des Ergebnisses nichts mehr falsch machen. Nur ist der Rechenaufwand sehr viel höher. Einfacher ist hier die Prüfung, ob die Tableaus für $\neg(\varphi\to\chi)$ sowie für $\neg(\chi\to\varphi)$ schließen. (Dies ist eine Übungsaufgabe.)

Übungen

[68] Zeigen Sie mit Hilfe eines Tableaus, dass $\mathtt{(p0\to p1)\to((p1\to p2)\to(p0\to p2))}$ eine Tautologie ist.

[69] Ist die Formel

$$((p1\to p3)\to\neg(\neg p7\wedge p1))\wedge(p1\vee\neg p3)$$

erfüllbar? Wenn ja, geben Sie eine Belegung an, die diese erfüllt. (Nehmen Sie dafür auf jeden Fall Tableaus, keine Wahrheitstafeln!)

[70] (Für Ehrgeizige.) Es sei die folgende Funktion „+” auf Wahrheitswerten definiert. $0 + 0 := 1 + 1 := 0$, $1 + 0 := 0 + 1 := 1$. (Dies ist das sogenannte *ausschließende Oder*.) Diese werde der Einfachheit halber durch das Symbol „+” bezeichnet. Versuchen Sie, Tableau-Regeln zu formulieren, die die Mengen $\Delta; \varphi + \chi$ bzw. $\Delta; \neg(\varphi + \chi)$ abbauen.

[71] Zeigen Sie, dass zwei Formeln φ und χ genau dann synonym sind, wenn sowohl $\varphi\wedge\neg\chi$ als auch $\chi\wedge\neg\varphi$ ein schließendes Tableau haben. Würde auch eine der beiden Bedingungen genügen? Wenn nein, warum nicht?

Kapitel 20

Folgerung

Einer der zentralen Begriffe der Logik ist der der *Folgerung*. Folgerung ist eine Beziehung zwischen Aussagen. Falls wir eine Aussage als wahr erkannt haben, können wir viele andere Aussagen auch als wahr erkennen, ohne dass wir noch mehr wissen müssen; denn ihre Wahrheit folgt aus dieser Formwl allein mittels der Bedeutungskonvention. Ist etwa die Aussage `p0` wahr, so auch die Aussage `p1∨p0`. Umgangssprachlich gesetzt: Ist der Satz S wahr, dann auch $/S\ \texttt{oder}\ T/$, egal, worum es sich bei T handelt. In der Literatur schreibt man das meist so auf

$$\frac{S}{\therefore S\ \texttt{oder}\ T} \tag{20.1}$$

Hierbei ist $\therefore$ symbolisch für den Schluss; wir könnten stattdessen auch hinschreiben „Also ist" oder „Dann ist". Bei den Formeln oberhalb des Strichs müssten wir ergänzen „Sei$\cdots$der Fall". Der Schluss liest sich also wie folgt.

Sei S. Also ist $S\,\texttt{oder}T$.

Die Sätze (oder Formeln) oberhalb des Strichs heißen **Prämissen**, die unterhalb des Strichs **Konklusion** oder **Folgerung**. Ein Schluss kann mehrere Prämissen haben, so etwa in

$$\frac{S \qquad \mathrm{T}}{\therefore S\ \texttt{und}\ T} \tag{20.2}$$

Es gibt aber nur eine einzige Konklusion. Doch davon später.

Zunächst will ich das Konzept des logischen Schlusses noch etwas konkretisieren. Ersetzen wir die Variable durch konkrete Aussagen:

$$\frac{\texttt{Es regnet.}}{\therefore \texttt{Es regnet oder es ist kalt.}} \tag{20.3}$$

Umgangssprachlich:

> Gesetzt, dass es regnet. Dann regnet es oder es ist kalt.

Wie gesagt, ist dies ein gültiger Schluss. Wir sagen dann, die zweite Aussage *folgt* aus der ersten. Dazu muss nur gelten, dass die zweite gilt, wenn die erste gilt. Die Schlussregel ist auch dann richtig, wenn die Bedingung gar nicht erfüllt ist. Dies ist der intuitiv schwierigste Teil bei logischen Schlüssen. Um zu wissen, ob (20.3) eine gültige Schlussregel ist, müssen wir nicht wissen, was passiert, wenn es nicht regnet. Dann ist ja die Prämisse falsch. Die Konklusion darf dann wahr oder falsch sein, es macht keinen Unterschied. Nur wenn die Prämisse wahr ist, muss auch die Konklusion wahr sein. Umgekehrt ist eine Schlussregel ungültig, wenn die Prämisse wahr sein kann, ohne dass es die Konklusion auch ist.

Zum Beispiel folgt /S `und` T/ *nicht* aus /S `oder` T/. Denn die letztere Aussage kann wahr sein, ohne dass die erste wahr ist (etwa, wenn nur S wahr ist, T aber nicht). Oder, um konkrete Aussagen zu nehmen, gesetzt, ich mag keine Torten aber Schinken. Dann ist sicher der Fall, dass ich Torten mag oder dass ich Schinken mag (das zweite gilt ja). Aber es keinesfalls richtig, dass ich Torten mag *und* dass ich Schinken mag (denn das erste ist falsch).

Definition 20.1 *Es sei Δ eine Menge von Formeln und φ eine Formel. φ* ***folgt aus*** *Δ, falls jede Belegung, die alle Formeln aus Δ erfüllt, auch φ erfüllt. Wir schreiben dies $\Delta \vDash \varphi$ und sagen, der* ***Schluss*** *von φ aus Δ sei* ***gültig****.*

Man beachte, dass „$\Delta \vDash \varphi$" eine Behauptung ist, nämlich die, dass φ aus Δ folgt.

Lemma 20.2 ① *Ist $\varphi \in \Delta$, so gilt $\Delta \vDash \varphi$.*

② *Ist $\Delta \vDash \chi$ und $\Delta \subseteq \Delta'$, so $\Delta' \vDash \chi$.*

③ *Ist $\Delta \vDash \chi$ für jedes $\chi \in \Sigma$, und ist $\Sigma \vDash \varphi$, so gilt auch $\Delta \vDash \varphi$.*

In diesem Zusammenhang gleich eine Vorausschau. Der Ausdruck „$\Delta \vdash \varphi$" (mit „$\vdash$" anstelle von „$\vDash$") wird auch eine **Regel** genannt. Wir sagen dann, die *Regel* $\Delta \vdash \varphi$ sei korrekt oder ein Schluss mit ihrer Hilfe sei gültig, wenn $\Delta \vDash \varphi$.

Als Beispiel gebe ich

(20.4) $\{\texttt{p0}, \texttt{p1}\} \vDash \texttt{p0}\wedge\texttt{p1}$

Dabei bedient man sich derselben Schreibweise, wie bei dem Tablaus: die Mengenklammern werden weggelassen, die Elemente voneinander durch Semikolon getrennt. Damit schreiben wir (20.4) auch

(20.5) $\texttt{p0}; \texttt{p1} \vDash \texttt{p0}\wedge\texttt{p1}$

Nach der Definition ist dieser Schluss gültig, wenn für alle Belegungen β gilt: ist $\beta(\texttt{p0}) = 1$ und $\beta(\texttt{p1}) = 1$, dann ist auch $\beta(\texttt{p0}\wedge\texttt{p1}) = 1$. In diesem Fall sieht man dies vielleicht mit bloßem Auge; wenn nicht, dann kann man einfach Wahrheitstafeln erstellen und damit alle Fälle durchgehen.

In der Definition einer Regel darf auch $\Delta = \varnothing$ sein. In diesem Fall ist $\Delta \vDash \varphi$ dann und nur dann, wenn φ unter jeder Belegung erfüllt ist, das heißt, wenn φ eine *Tautologie* ist. Wir schreiben dann

(20.6) $\vDash \varphi$

anstelle von $\varnothing \vDash \varphi$.

Lemma 20.3 *Enthält Δ eine Kontradiktion oder ist φ eine Tautologie, dann ist $\Delta \vDash \varphi$. Ist Δ nicht erfüllbar, so ist ebenfalls $\Delta \vDash \varphi$.*

(20.5) zeigt, dass die folgende Regel korrekt ist:

(20.7) $\texttt{p0}; \texttt{p1} \vdash \texttt{p0}\wedge\texttt{p1}$

Noch einmal: die Tatsache, dass $\Delta \vDash \varphi$, ist die Begründung dafür, dass die *Regel* $\Delta \vdash \varphi$ korrekt ist.

Eine andere Regel ist der sogenannte **Modus Ponens**:

(20.8) $\texttt{p0}; \texttt{p0}\to\texttt{p1} \vdash \texttt{p1}$

Denn ist β eine Belegung, die $\texttt{p0}$ erfüllt sowie $\texttt{p0}\to\texttt{p1}$, so kann nicht $\beta(\texttt{p1}) = 0$ sein, denn dann wäre ja $\beta(\texttt{p0}\to\texttt{p1}) = 0$.

Noch ein Beispiel. Die folgende Regel ist korrekt.

(20.9) $\texttt{(p0}\to\texttt{p1)}\to\texttt{p0} \vdash \texttt{p0}$

Dazu müssen wir zeigen, dass

(20.10) $\texttt{(p0}\to\texttt{p1)}\to\texttt{p0} \vDash \texttt{p0}$

Sei dazu $\beta(\texttt{(p0}\to\texttt{p1)}\to\texttt{p0}) = 1$. Dann ist $\beta(\texttt{p0}) = 1$ (und wir sind fertig) oder $\beta(\texttt{p0}) = 0$. Im zweiten Fall ist $\beta(\texttt{p0}\to\texttt{p1}) = 1$ und $\beta(\texttt{(p0}\to\texttt{p1)}\to\texttt{p0}) = 0$, und wir sind ebenfalls fertig.

Zwischen den Gleichungen und den Schlüssen besteht ein enger Zusammenhang. Offenbar ist $\Delta \vDash \varphi$ genau dann, wenn

(20.11) $\{\beta : \text{für alle } \delta \in \Delta\colon \beta(\delta) = 1\} \subseteq [\varphi]$

Das kann ich auch so schreiben:

$$\bigcap_{\delta\in\Delta}[\delta] \subseteq [\varphi] \tag{20.12}$$

Insbesondere für $\Delta = \{\chi\}$ bekommen wir $\Delta \vDash \varphi$ genau dann, wenn $[\chi] \subseteq [\varphi]$. Nun ist $[\chi] = [\varphi]$ äquivalent mit $[\chi] \subseteq [\varphi]$ *und* $[\varphi] \subseteq [\chi]$. Daraus bekommen wir folgenden Satz.

Proposition 20.4 *Ist $\varphi \approx \chi$, so ist $\varphi \vDash \chi$ (und auch $\chi \vDash \varphi$). Ist sowohl $\varphi \vDash \chi$ als auch $\chi \vDash \varphi$, so ist $\varphi \approx \chi$.*

Auf diese Weise lässt sich der Kalkül für $\approx$ auf den Kalkül für $\vDash$ zurückführen. $\varphi \approx \chi$ wird gezeigt, indem man $\varphi \vDash \chi$ und $\chi \vDash \varphi$ zeigt. Ein umgekehrte Reduktion ist auch möglich.

Proposition 20.5 *Genau dann ist $\varphi \vDash \chi$, wenn $\varphi\vee\chi \approx \chi$.*

Beweis. Es sei $\varphi \vDash \chi$. Dann ist auch $\varphi\vee\chi \vDash \chi$. Denn ist $\beta(\varphi\vee\chi) = 1$, so ist entweder $\beta(\varphi) = 1$, und dann $\beta(\chi) = 1$ nach Annahme, oder aber $\beta(\chi) = 1$ und die Behauptung gilt. Es gilt gewiss $\chi \vDash \varphi\vee\chi$. Also haben wir $\varphi\vee\chi \vDash \chi$ und $\chi \vDash \varphi\vee\chi$. Nach dem vorigen Satz ist dann $\varphi\vee\chi \approx \chi$. Das zeigt die Richtung von links nach rechts. Die Umkehrung ist wie folgt. Nehmen wir an, dass $\varphi\vee\chi \approx \chi$, so ist wieder nach dem vorigen Lemma $\varphi\vee\chi \vDash \chi$. Sei nun $\beta(\varphi) = 1$. Dann ist $\beta(\varphi\vee\chi) = 1$, und deswegen $\beta(\chi) = 1$. Das bedeutet, $\varphi \vDash \chi$. ⊣

Eine ganz wichtige Regel ist der sogenannte Modus Ponens. Der Modus Ponens besagt, dass wir aus einer Formel φ in Gegenwart der Formel $\varphi\to\chi$ die Formel χ schließen dürfen. Dies wird im Allgemeinen so wiedergegeben.

$$\varphi; \varphi\to\chi \vdash \chi \tag{20.13}$$

In dieser Formulierung sind aber die φ und χ nur schematische Buchstaben: wir dürfen an ihre Stellen jede beliebige Formel setzen, erst dann bekommen wir eine Regel. Das bedeutet, dass (20.14) ein Regelschema ist, das folgende hingegegen eine Instanz davon:

$$\texttt{p0}\to\neg\texttt{p2}; (\texttt{p0}\to\neg\texttt{p2})\to(\texttt{p1}\vee\texttt{p0}) \vdash \texttt{p1}\vee\texttt{p0} \tag{20.14}$$

Ebenso ist (20.8) lediglich eine Instanz des Schemas. Dennoch hat man den Verdacht, dass (20.8) schon alleine der Modus Ponens ist. Dies ist in einem gewissen Sinne richtig: denn alle Instanzen lassen sich aus (20.8) durch sogenannte

Substitution gewinnen. Ich habe Substitutionen schon vorher verwendet; ich habe $[\mathtt{p0}\vee\mathtt{p1}/\mathtt{p2}]$ geschrieben für die Operation, welche $\mathtt{p2}$ durch $\mathtt{p0}\vee\mathtt{p1}$ ersetzt. Solche Ersetzungen kann man auch simultan über mehrere Variablen definieren, etwa $[\mathtt{p0}\vee\mathtt{p1}/\mathtt{p2}, \neg\mathtt{p1}/\neg\mathtt{p1}]$. Technisch ist es oft einfacher, der Ersetzungsoperation einen Namen zu geben. Gleichzeitig werden wir ihren Effekt durch eine induktive Definition beschreiben.

Definition 20.6 *Eine **Substitution** ist eine Abbildung σ von den Variablen in die Menge der Aussagen. Ist σ eine Substitution, so setzen wir*

$$
\begin{array}{ll}
h_\sigma(p\vec{x}) & := \sigma(p\vec{x}) \\
h_\sigma((\neg\chi)) & := (\neg h_\sigma(\chi)) \\
h_\sigma((\varphi\wedge\chi)) & := (h_\sigma(\varphi)\wedge h_\sigma(\chi)) \\
h_\sigma((\varphi\vee\chi)) & := (h_\sigma(\varphi)\vee h_\sigma(\chi)) \\
h_\sigma((\varphi\to\chi)) & := (h_\sigma(\varphi)\to h_\sigma(\chi))
\end{array}
\tag{20.15}
$$

(Ich habe hier wieder die alte, korrekte Klammerschreibweise genommen. Dies ist die einzige Art, Risiken zu vermeiden. Die Klammersparkonventionen dürfen erst nachträglich eingesetzt werden. Nehmen wir als Beispiel die Formel $\mathtt{p0}\wedge\mathtt{p1}$. Falls $\sigma(\mathtt{p0}) = \mathtt{p1}\vee\mathtt{p2}$, so bekommen wir $\mathtt{p1}\vee\mathtt{p2}\wedge\mathtt{p1}$, welches aber nicht äquivalent ist zu $(\mathtt{p1}\vee\mathtt{p2})\wedge\mathtt{p2}$. Es muss mindestens gesichert sein, dass die ersetzenden Formeln, wenn sie keine Variable sind, in Klammern gesetzt sind. Im Beispiel muss $\sigma(\mathtt{p0}) = (\mathtt{p1}\vee\mathtt{p2})$ sein.) Ich erinnere noch einmal an die in Teil I eingeführte Notation: Ist $f : M \to N$ eine Funktion und $H \subseteq M$, so bezeichnet $f[H]$ die Menge $\{f(x) : x \in H\}$. Genauso bezeichnet jetzt $h_\sigma[\Delta]$ die Menge $\{h_\sigma(\delta) : \delta \in \Delta\}$.

Satz 20.7 *Ist $\Delta \vDash \varphi$, so auch $h_\sigma[\Delta] \vDash h_\sigma(\varphi)$.*

Daraus ziehen wir die erste Folgerung. Der Schluss (20.8) ist gültig. Sei σ eine beliebige Substitution mit $\sigma(\mathtt{p0}) = \varphi$ und $\sigma(\mathtt{p1}) = \chi$. Dann ist nach Satz 20.7 auch die folgende Regel gültig:

$$\varphi; \varphi\to\chi \vdash \chi \tag{20.16}$$

Zum Beispiel $\sigma(\mathtt{p0}) := \mathtt{p0}\to\neg\mathtt{p2}$, $\sigma(\mathtt{p1}) := \mathtt{p1}\vee\mathtt{p0}$ ergibt gerade (20.14). Denn es ist $h_\sigma(\mathtt{p0}\to\mathtt{p1}) = h_\sigma(\mathtt{p0})\to h_\sigma(\mathtt{p1}) = (\mathtt{p0}\to\neg\mathtt{p2})\to\mathtt{p1}\vee\mathtt{p0}$.

Zum Schluss erwähne ich noch eine sehr nützliche Tatsache.

Satz 20.8 (Deduktionstheorem) *Es ist $\Delta; \varphi \vDash \chi$ genau dann, wenn $\Delta \vDash \varphi\to\chi$.*

Beweis. ($\Rightarrow$) Gelte $\Delta; \varphi \vDash \chi$, und sei β eine Belegung, die Δ erfüllt. (Fall 1) $\beta(\varphi) = 1$. Dann ist nach Voraussetzung $\beta(\chi) = 1$, also ist $\beta(\varphi\to\chi) = 1$. (Fall 2) $\beta(\varphi) = 0$. Dann ist $\beta(\varphi\to\chi) = 1$. Also gilt in jedem Fall $\Delta \vDash \varphi\to\chi$. ($\Leftarrow$) Sei $\Delta \vDash \varphi\to\chi$ und β eine Belegung, die Δ und φ erfüllt. Dann erfüllt β nach Voraussetzung Δ und somit $\varphi\to\chi$. Da jetzt $\beta(\varphi) = 1$ und $\beta(\varphi\to\chi) = 1$, so ist $\beta(\chi) = 1$. Also gilt $\Delta; \varphi \vDash \chi$. ⊣

Sehen wir uns das für unsere Regeln an. Wir haben

(20.17) $\quad$ `p0; p1` $\vDash$ `p0∧p1`

Wenden wir das Deduktionstheorem einmal an, so bekommen wir

(20.18) $\quad$ `p0` $\vDash$ `p1→p0∧p1`

Da die Reihenfolge der Formeln links von $\vDash$ keine Rolle spielt, können wir auch dies ableiten:

(20.19) $\quad$ `p1` $\vDash$ `p0→p0∧p1`

Wenden wir das Deduktionstheorem zweimal auf (20.17) an, so bekommen wir

(20.20) $\quad$ $\vDash$ `p0→p1→p0∧p1`

oder auch

(20.21) $\quad$ $\vDash$ `p1→p0→p0∧p1`

je nachdem, wie wir verfahren.

Man bedenke, dass der Pfeil rechtsassoziativ ist und schwächer bindet als die Konjunktion. Die Zeichenkette in (20.20) ist also eine Kurzschreibweise für

(20.22) $\quad$ $\vDash$ `(p0→(p1→(p0∧p1)))`

Wir haben auch

(20.23) $\quad$ `p0; p0→p1` $\vDash$ `p1`

Wenden wir hier das Deduktionstheorem zweimal an, bekommen wir

(20.24) $\quad$ $\vDash$ `p0→(p0→p1)→p1`

Das Deduktionstheorem ist manchmal für sehr schnelle Beweise gut. Ich zeige zum Beispiel

(20.25) $\quad$ `p0→p1→p2` $\vDash$ `p1→p0→p2`

Und zwar beginnen wir mit

(20.26) $\texttt{p0} \vDash \texttt{p0}$

Wir wenden die Substitution $\texttt{p0} \mapsto \texttt{p0→p1→p2}$ and und bekommen:

(20.27) $\texttt{p0→p1→p2} \vDash \texttt{p0→p1→p2}$

Zweimaliges Anwenden des Deduktionstheorems ergibt

(20.28) $\texttt{p0→p1→p2; p1; p0} \vDash \texttt{p2}$

Man beachte, dass es links nicht auf Reihenfolge ankommt (links vom $\vDash$ steht ja technisch gesehen eine Menge). Wir wenden das DT also erst einmal auf $\texttt{p0}$ an:

(20.29) $\texttt{p0→p1→p2; p1} \vDash \texttt{p0→p2}$

Dann wenden wir es auf $\texttt{p1}$ an:

(20.30) $\texttt{p0→p1→p2} \vDash \texttt{p1→p0→p2}$

Das ist das gewünschte Ergebnis.

Ich komme zum Schluss noch auf die Bedeutung des Ganzen zu sprechen. Im Augenblick scheint alles etwas zu verschwimmen: *Folgerung*, *Regel*, *Schluss* erscheinen einem ununterscheidbar. Das Deduktionstheorem macht alles nur noch schlimmer. Was ist der Sinn des Ganzen? Dazu sei zunächst einmal unterschieden zwischen einem Schluss und einer Regel. Die Regel des Modus Ponens ist nichts weiter als eine Art Lizenz zum Schließen. Sie ist gewissermaßen ein Gesetz, das in gegebenen Fällen zur Anwendung kommen kann. Habe ich gesichert, dass gewisse Sätze wahr sind, die die Form φ→χ und φ haben, so darf ich mit der Regel des MP jetzt χ schließen. Ich darf so schließen, ich muss es aber nicht. Weil die Regel allerdings auch korrekt ist, habe ich zusätzlich noch die Gewissheit, dass die Folgerung, also der Satz χ, wahr ist. Regeln sind also zeitlose Gebilde, reine Syntax, während sich Schlüsse in der Zeit vollziehen. *Wir schließen mit Hilfe von Regeln.* Natürlich können wir auch ohne Regeln schließen, aber erst die Systematisierung der Schlüsse (und die damit einhergehende Prüfung der Regeln auf Korrektheit) sichert uns, dass wir *richtig*, das heißt *korrekt*, schließen.

Das Deduktionstheorem kann man erst verstehen, wenn man zwischen einem Satz und der Behauptung seiner Wahrheit unterscheiden lernt. Ein Satz ist zunächst einmal nur ein syntaktisches Gebilde. Erst wenn ich ihn in einer gewissen

Weise ausspreche, behaupte ich seine Wahrheit (ich vollziehe einen assertorischen Sprechakt). Mittels der ußerung einer Aussage der Form `p0→p1` behauptet man die Wahrheit von `p0→p1`, nicht mehr und nicht weniger. Die Wahrheit der darin enthaltenen Teilsätze wird nicht behauptet. Genauso ist die Behauptung von `p0∧p1` zunächst eine einfache Behauptung (ein einziger Sprechakt). Dass man stattdessen auch erst einmal `p0` und dann — getrennt davon — hätte `p1` behaupten kann, ist davon unabhängig. Es liegt an der Semantik der Konjunktion, dass dem so ist.

Was aber steckt hinter `p0 ⊨ p1→p0` bzw. `p0 ⊢ p1→p0`? Wenden wir dem zweiten zu: `p0 ⊢ p1→p0` besagt, dass ich `p1→p0` *behaupten* darf, wenn ich `p0` als wahr erkannt habe. Es ist also eine Art Gesprächsregel: Hat man die Aussage `p0` als wahr erkannt, so darf man auch `p1→p0` unter Berufung auf diese Regel behaupten. Man beachte, dass es sich jetzt um *zwei* Behauptungen handelt im Gegensatz zu `p0→(p1→p0)`, das jeweils nur als Ganzes behauptet werden kann. Die Begründung dafür, dass der Übergang von der Wahrheit von `p0` zu der Behauptung, dass `p1→p0`, problemlos ist, liegt in dem *Sachverhalt*, dass `p0 ⊨ p1→p0`.

Übungen

[72] Modus Ponens ist die Regel `p0; p0→p1 ⊢ p1`. Zeigen Sie, dass die Regel „MP unter einer Bedingung" korrekt ist:

(20.31) `p0→p1; p0→p1→p2 ⊢ p0→p2`

Achtung: Sie müssen die Klammerkonventionen beachten!

[73] Leiten Sie aus der Korrektheit von (20.31) ab, dass die folgenden Formeln Tautologien sind:

(20.32) `(p0→p1→p2)→(p0→p1)→p0→p2`
`(p0→p1)→(p0→p1→p2)→p0→p2`

[74] Zeigen Sie, dass folgende Regel korrekt ist: `¬p0→p0 ⊢ p0`.

[75] Überlegen Sie, dass aus der vorigen Behauptung folgt, dass die folgende Formel eine Tautologie ist: `(¬p0→p0)→p0`. (*Anmerkung.* Wenn man noch überlegt, dass `¬p0` und `p0→f` synonym sind, so können wir daraus auch noch ableiten, dass

`((p0→f)→p0)→p0`

eine Tautologie ist. Das ist eine Variante von Peirce's Formel, `((p0→p1)→p0)→p0`.)

Kapitel 21

Ein Deduktionskalkül

In dieser Vorlesung werde ich einen Kalkül einführen, mit dem man das Schließen axiomatisieren kann. Dabei ist eine Axiomatisierung ein Verfahren, das rein mechanisch und ohne Zuhilfenahme nichtsyntaktischer Eigenschaften, wie etwa Semantik, entscheiden lässt, ob eine Formel aus einer Menge von Formeln folgt oder nicht. Solche Kalküle gibt es für die Aussagenlogik und für die Prädikatenlogik, die ich im Anschluss einführen werde. Die Kalküle besitzen immer eine Menge von sogenannten **Axiomen**, welche spezielle Formeln sind, und **Regeln**, welche erlauben, Formeln aus Formeln herzuleiten. Der Kalkül erzeugt so eine (meist unendliche) Menge von Formeln, nämlich alle diejenigen, die entweder Axiome sind oder mit Hilfe von Regeln aus Axiomen abgeleitet werden können, wobei die Länge der Ableitung hier keine Rolle spielen soll. (Natürlich sind Beschränkungen in der Länge in der Praxis relevant, aber hier geht es um den theoretischen Begriff der Ableitung.)

Es gibt sehr viele solche Kalküle. Der bekannteste ist der sogenannte **Hilbert Kalkül**. Dieser Kalkül besitzt sehr viele Axiome und nur zwei Regeln. Ich präsentiere den Kalkül in zwei Fassungen, die im Wesentlichen gleichartig sind. Sie dienen allerdings verschiedenen Zwecken. Die erste Fassung ist ein Kalkül, um Tautologien herzuleiten. Die zweite ist eine Erweiterung und erlaubt, das Folgern aus Prämissen (also das Schließen, wie in der vorangegangenen Vorlesung vorgestellt) zu charakterisieren. Mit anderen Worten, der erste Kalkül erlaubt uns zu sehen, ob eine gegebene Formel allgemeingültig ist, das heißt, ob sie eine Tautologie ist. Die zweite Fassung erlaubt uns zu sagen, ob eine Formel φ aus einer Formelmenge Δ folgt. Da wir für das Schließen ja das Deduktionstheorem haben, ist der erste Kalkül eigentlich ausreichend. Wollen wir zum Beispiel wissen, ob wir φ aus $\{\delta_0, \delta_1\}$ herleiten können, das heißt, ob $\delta_0; \delta_1 \vDash \varphi$, so genügt zu zeigen, dass $\delta_0 \to \delta_1 \to \varphi$ (oder auch $\delta_1 \to \delta_0 \to \varphi$) eine Tautologie ist. Trotzdem wird es sich lohnen, auch den zweiten Kalkül gesondert vorzustellen.

Beginnen wir also mit dem ersten Kalkül. Zunächst präsentiere ich eine Liste von Formeln.

(21.1)

(pb)	`p0→p1→p0`
(fd)	`(p0→p1→p2)→(p0→p1)→p0→p2`
(pc)	`((p0→p1)→p0)→p0`
(l∧)	`p0→p1→p0∧p1`
(r∧)	`p0→p1→p1∧p0`
(∧l)	`p0∧p1→p0`
(∧r)	`p0∧p1→p1`
(l∨)	`(p0→p2)→(p1→p2)→(p0∨p1)→p2`
(r∨)	`(p0→p2)→(p1→p2)→(p1∨p0)→p2`
(∨l)	`p0→(p0∨p1)`
(∨r)	`p1→(p0∨p1)`
(e¬)	`(p0→f)→¬p0`
(¬e)	`¬p0→p0→f`
(dn)	`¬¬p0→p0`
(efq)	`f→p0`
(t)	`w`

Dies ist eine endliche Liste, bestehend aus genau 16 Formeln. Alle diese Formeln heißen **Axiome** des Kalküls. Eine Formel ist also genau dann Axiom, wenn sie in der Liste vorkommt. Axiome sind Formeln, die ohne weiteres Zutun als gültig anerkannt sind. Regeln sind dagegen Prozeduren, die besagen, wie wir aus gültigen Formeln neue gültige Formeln schließen können. Die Regeln dieses Kalküls sind Substitution und Modus Ponens. Die Substitutionsregel erlaubt, von einer Formel φ zu der Formel $h_\sigma(\varphi)$ überzugehen. All dies ist in der folgenden Definition untergebracht.

Definition 21.1 *Eine* ***H-Ableitung*** *von φ ist eine Folge $\langle \delta_i : i < n+1 \rangle$ derart, dass $\delta_n = \varphi$ und für jedes $i < n+1$ eines der Folgenden gilt.*

① *δ_i ist ein Axiom.*

② *Es existiert $j < i$ und eine Substitution σ mit $h_\sigma(\delta_j) = \delta_i$.*

③ *Es existieren $j, k < i$ derart, dass $\delta_k = \delta_j \to \delta_i$ ist.*

Wir sagen, φ sei ***H-ableitbar****, in Zeichen $\vdash \varphi$, wenn es eine H-Ableitung von φ gibt.*

(Hierbei steht das „H" für *Hilbert*.) Das Folgende ist zum Beispiel eine H-Ableitung von $\texttt{p0→w}$.

(21.2) $\langle \texttt{p0→p1→p0}, \texttt{w→p0→w}, \texttt{w}, \texttt{p0→w} \rangle$

Da solche Sequenzen etwas mühsam zu lesen sind, reichert man sie mit etwas Information an.

(21.3)

1.	(pb)	$\texttt{p0→p1→p0}$
2.	Sub:1	$\texttt{w→p0→w}$
3.	(t)	$\texttt{w}$
4.	MP:2,3	$\texttt{p0→w}$

Die Zeilen sind durchnummeriert. Ist die Zeile ein Axiom, so schreibe ich den Namen links hin. Ist eine Regel angewendet worden, so steht im Falle der Substitution „Sub:" gefolgt von der Nummer der Zeilen, auf die die Substitution angewendet wurde, sowie im Fall des Modus Ponens „MP:" gefolgt von den Nummern der zwei Zeilen, auf die Modus Ponens angewendet wurde. Im oberen Beispiel entsteht die zweite Zeile also aus der ersten durch eine Substitution; und zwar nehme man $\texttt{p0} \mapsto \texttt{w}, \texttt{p1} \mapsto \texttt{p0}$. Die vierte entsteht aus 2 und 3 durch Modus Ponens.

Hier ist eine H-Ableitung von $\texttt{p0→p0}$.

(21.4)

1.	(fd)	$\texttt{(p0→p1→p2)→(p0→p1)→p0→p2}$
2.	Sub:1	$\texttt{(p0→(p0→p0)→p0)→(p0→p0→p0)→p0→p0}$
3.	(pb)	$\texttt{p0→p1→p0}$
4.	Sub:3	$\texttt{p0→(p0→p0)→p0}$
5.	MP:2,4	$\texttt{(p0→p0→p0)→p0→p0}$
6.	Sub:3	$\texttt{p0→p0→p0}$
7.	MP:5,6	$\texttt{p0→p0}$

Die Substitution zur 2. Zeile ist $\texttt{p0} \mapsto \texttt{p0}$, $\texttt{p1} \mapsto \texttt{(p0→p0)}$ und $\texttt{p2} \mapsto \texttt{p0}$. Zweimal wird eine Substitution auf Zeile 3 angewendet. (Natürlich hätten wir $\texttt{p0→p0}$ in unseren Kalkül als Axiom aufnehmen können, und manche tun dies der Einfachheit halber. Auf der anderen Seite ist es formal nicht nötig, und mir kommt es nicht auf die konkrete Wahl der Axiome an, nur auf die Technik der Axiomatisierung und wie man mit ihr umgeht.)

Satz 21.2 *Genau dann ist φ H-ableitbar, wenn $\vDash \varphi$, dh wenn φ eine Tautologie ist.*

Dieser Satz zerfällt in zwei Teile: die Implikation von links nach rechts heißt die **Korrektheit** des Kalküls; der Kalkül ist korrekt, weil er nur Tautologien abzuleiten gestattet. Die Umkehrung heißt die **Vollständigkeit**. Alles, was eine Tautologie ist, ist auch H-ableitbar.

Für die Korrektheit müssen wir nicht sehr viel tun. Zunächst müssen wir nachrechnen, dass alle Formeln oben (die Axiome) auch wirklich Tautologien sind. Und dann müssen wir zeigen, dass jeder Schritt in einer Ableitung von Tautologien zu Tautologien führt. Dazu sei zum Beispiel δ_j eine Tautologie; ist dann σ eine Substitution, dann ist auch $h_\sigma(\delta_j)$ eine Tautologie. Ist nämlich β eine Belegung, so definiere $\gamma(p) := \beta(h_\sigma(p))$. Dann kann man zeigen, dass $\beta(h_\sigma(\delta_j)) = \gamma(\delta_j)$. Und weil δ_j Tautologie ist, ist $\gamma(\delta_j) = 1$. Nun noch zum zweiten Teil: ist δ_j eine Tautologie und $\delta_j \rightarrow \delta_i$ eine Tautologie, so auch δ_i. Denn sei β eine Belegung. Nach Voraussetzung ist $\beta(\delta_j) = 1$ sowie $\beta(\delta_j \rightarrow \delta_i) = 1$. Also ist dann auch $\beta(\delta_i) = 1$. Die Vollständigkeit wird aus Satz 21.10 folgen.

Wir wollen nun auch noch einen Kalkül definieren, mit dem man feststellen kann, ob $\Delta \vDash \varphi$. Bevor wir dies tun, will ich eine Vorbemerkung machen. In vielen Büchern wird der Hilbert-Kalkül anders gefasst. Die Axiome werden lediglich schematisch angegeben, etwa ist (pb) in solchen Kalkülen das Schema $\varphi \rightarrow \chi \rightarrow \varphi$. Dies Schema steht für unendlich viele Axiome. Als Schlussregel hat man aber nur noch Modus Ponens. Mit einem solchen Kalkül kann man aber auch alle Formeln ableiten, die H-ableitbar sind. Zunächst einmal kann man sicher alle Substitutionsinstanzen von Axiomen ableiten. Der H-Kalkül erlaubt aber Substitution auf bereits abgeleitete Formeln. Dies ist aber entbehrlich. Dazu zeigt man die sogenannte Vorverlegung der Substitution: alles, was H-ableitbar ist, ist auch aus Substitutionen von Axiomen ohne den Gebrauch von Substitution, also nur mit Modus Ponens, ableitbar. Ist nämlich δ_i aus δ und $\delta \rightarrow \eta$ abgeleitet worden und anschließend $h_\sigma(\chi)$, so genügt es, zunächst auf δ und $\delta \rightarrow \delta_i$ die Substitution anzuwenden:

$$
\begin{array}{c}
\cdots \delta_j, \cdots \delta_j \rightarrow \delta_i, \cdots \delta_i, \quad h_\sigma(\delta_i) \\
\downarrow \\
\cdots \delta_j, \quad h_\sigma(\delta_j) \cdots \delta_j \rightarrow \delta_i, \quad h_\sigma(\delta_j \rightarrow \delta_i) \cdots h_\sigma(\delta_i)
\end{array}
\tag{21.5}
$$

Da $h_\sigma(\delta \rightarrow \eta) = h_\sigma(\delta) \rightarrow h_\sigma(\eta)$, ist das Untere auch ein Beweis. Die Substitution ist nun vorverlegt. Anstatt auf die $i$te Formel wie im oberen Beweis wird sie auf die $j$te bzw. $k$te Formel angewendet, wobei nach Annahme $j, k < i$. Was aber, wenn zwei Substitutionen aufeinandertreffen? Was ist mit $h_\tau(h_\sigma(\delta_i))$? Hier gilt, dass die zweifache Substitution durch eine einzige ersetzt werden kann, die wir mit $\tau \circ \sigma$

bezeichnen. Diese ist wie folgt definiert:

$$\tau \circ \sigma(p) := h_\tau(\sigma(p)) \qquad (21.6)$$

Mit dieser Vorbemerkung steigen wir nun in die Definition des neuen Kalküls ein. Dieser ist wie folgt definiert.

Definition 21.3 *Eine* ***C-Ableitung*** *von φ aus Σ ist eine Folge $\langle \delta_i : i < n+1 \rangle$ derart, dass $\delta_n = \varphi$ und für jedes $i < n+1$ eines der Folgenden gilt.*

① $\delta_i \in \Sigma$.

② *δ_i ist eine Substitutionsinstanz eines Axioms.*

③ *Es existieren $j, k < i$ derart, dass $\delta_k = \delta_j$→δ_i ist.*

Wir sagen, φ sei aus Σ ***C-ableitbar****, in Zeichen $\Sigma \vdash \varphi$, wenn es eine C-Ableitung von φ aus Σ gibt.*

Die Notation von C-Ableitungen ist wie die von H-Ableitungen. Es gibt nur noch eine neue Annotation, nämlich Ann für Annahme. Ich zeige zum Beispiel `p0→p1`; `¬p1` ⊢ `¬p0`.

(21.7)

1.	Ann	`p0→p1`
2.	Ann	`¬p1`
3.	(¬e)	`¬p1→p1→f`
4.	MP:2,3	`p1→f`
5.	(fd)	`(p0→p1→f)→(p0→p1)→(p0→f)`
6.	(pb)	`(p1→f)→p0→p1→f`
7.	MP:4,6	`p0→p1→f`
8.	MP:5,7	`(p0→p1)→(p0→f)`
9.	MP:1,8	`p0→f`
10.	(e¬)	`(p0→f)→¬p1`
11.	MP:9,10	`¬p1`

Ist auf φ aus $\varnothing$ C-herleitbar, so ist φ auch H-herleitbar. Denn jede C-Ableitung aus $\varnothing$ ist auch eine H-Ableitung. Da aber nicht jede H-Ableitung von φ auch eine C-Ableitung von φ aus $\varnothing$ ist, ist die folgende Tatsache schon nicht banal, folgt aber aus der Vorverlegung der Substitution, wie wir sie oben diskutiert haben.

Satz 21.4 *Genau dann ist φ H-ableitbar, wenn φ aus $\varnothing$ C-ableitbar ist.*

Es fallen also die Begriffe „C-ableitbar aus $\varnothing$" und „H-ableitbar" zusammen und wir sprechen fortan nur noch von „ableitbar" und unterscheiden diese Begriffe auch nicht symbolisch.

Zunächst einige leichte Folgerungen.

Satz 21.5 *1. Ist $\delta \in \Delta$, so gilt $\Delta \vdash \delta$.*

2. Ist $\Delta \vdash \chi$ und $\Delta \subseteq \Delta'$, so $\Delta' \vdash \chi$.

3. Ist $\Delta \vdash \eta$ und $\Delta; \eta \vdash \chi$, so ist $\Delta \vdash \chi$.

Beweis. Zu (1). Die Folge $\langle \delta \rangle$ ist schon eine C-Ableitung. Zu (2). Ist $\langle \delta_i : i < n \rangle$ eine Ableitung von χ aus Δ, so auch eine Ableitung von χ aus Δ'. Zu (3). Es sei $\langle \delta_i : i < p \rangle$ eine Ableitung von η aus Δ, $\langle \theta_j : j < q \rangle$ eine Ableitung von χ aus $\Delta; \eta$. Dann schreibe die folgende Sequenz hin.

(21.8) $$\delta_0, \delta_1, \cdots, \delta_{p-1}, \theta_0, \cdots, \theta_{q-1}$$

Dies ist (fast) eine C-Ableitung von χ aus Δ. Sei eine Formel in dieser Folge gegeben. Dann ist sie entweder eine Formel δ_i und somit entweder in Δ oder eine Instanz eines Axioms oder geht durch aus Formeln δ_j, δ_k, $j, k < i$, durch Anwendung von MP hervor. Oder sie ist eine Formel der Form θ_i, und dann ist sie gleich η oder eine Instanz eines Axioms oder sie geht aus Formeln θ_j, θ_k, $j, k < i$, durch MP hervor. Wir lassen nun alle θ_c weg, für die $\theta_c = \eta$. Die entstehende Folge ist immer noch eine C-Ableitung. Ist nämlich θ_i nicht Instanz eines Axioms oder aus Δ, so existieren $j, k < i$, sodass $\theta_i = \theta_j \rightarrow \theta_k$. Ist nun $\theta_j = \eta$ oder $\theta_k = \eta$, so sind diese jetzt weggefallen. Dann wählen wir einfach δ_{p-1} an ihrer Stelle, das ja nach wie vor in der Ableitung steht und vor θ_k vorkommt. ⊣

Daraus leite ich noch eine leicht andere Version ab.

Satz 21.6 *1. Ist $\Delta \vdash \theta$ und $\theta \vdash \chi$, so $\Delta \vdash \chi$.*

2. Ist $\Delta \vdash \theta$ für jedes $\theta \in \Theta$, und ist $\Theta \vdash \chi$, so ist $\Delta \vdash \chi$.

Beweis. Zu (1). Sei $\Delta \vdash \theta$ und $\theta \vdash \chi$. Dann ist nach Satz 21.5 (2) $\Delta; \theta \vdash \chi$, und wegen Satz 21.5 (3) ist $\Delta \vdash \chi$. Zu (2). Sei $\Theta = \{\theta_i : i < n\}$. Setze $\Theta' := \{\theta_i : i < n - 1\}$. Es ist $\Delta; \Theta' \vdash \theta_{n-1}$, nach Annahme, und $\Delta; \Theta'; \theta_{n-1} \vdash \chi$. Mit Satz 21.5 (3) bekommen wir $\Delta; \Theta' \vdash \chi$. Setze nun $\Theta'' := \{\theta_i : i < n - 2\}$. Wir haben $\Delta; \Theta'' \vdash \theta_{n-2}$ und $\Delta; \Theta''; \theta_{n-2} \vdash \chi$. Wiederum gilt nach Satz 21.5 (3), dass $\Delta; \Theta'' \vdash \chi$. Und so weiter, bis wir $\Delta \vdash \chi$ bekommen. ⊣

Wir können im Übrigen für den C-Kalkül auch das Deduktionstheorem beweisen.

Satz 21.7 *Genau dann ist φ aus $\Delta; \chi$ C-ableitbar, wenn $\chi \to \varphi$ aus Δ C-ableitbar ist.*

Beweis. ($\Leftarrow$) Es sei $\langle \delta_i : i < n+1 \rangle$ eine C-Ableitung von $\chi \to \varphi$ aus Δ. Dann ist das Folgende eine C-Ableitung von φ aus $\Delta; \chi$:

$$\delta_0, \delta_1, \cdots, \delta_n (= \chi \to \varphi), \chi, \varphi \tag{21.9}$$

($\Rightarrow$) Es sei eine C-Ableitung $\langle \delta_i : i < n+1 \rangle$ von φ aus $\Delta; \chi$ gegeben. Wir definieren eine C-Ableitung wie folgt. Für jedes $i < n+1$ setzen wir eine Folge von Formeln an die Stelle von δ_i. Und zwar:

- Ist $\delta_i \in \Delta$, so setzen wir die Folge $\langle \delta_i, \delta_i \to \chi \to \delta_i, \chi \to \delta_i \rangle$ ein.
- Ist $\delta_j = \chi$, so setzen wir die Folge

  $$\begin{array}{l} \langle \chi \to (\chi \to \chi) \to \chi, \\ (\chi \to (\chi \to \chi) \to \chi) \to (\chi \to \chi \to \chi) \to \chi \to \chi, \\ (\chi \to \chi \to \chi) \to \chi \to \chi, \\ (\chi \to \chi \to \chi), \\ \chi \to \chi \rangle \end{array}$$

  ein.
- Ist δ_i aus δ_j und $\delta_k = \delta_j \to \delta_i$ mit Modus Ponens entstanden, etwa $j < k$:

  $$\cdots \delta_j \cdots \delta_j \to \delta_i \cdots \delta_i \cdots$$

  So steht anstelle von δ_j nunmehr eine Folge, deren letzte Formel $\chi \to \delta_j$ ist anstelle von $\delta_j \to \delta_i$ eine Folge, deren letzte Formel $\chi \to \delta_j \to \delta_i$ ist. Jetzt gehen wir so weiter:

  $$\cdots \chi \to \delta_j \cdots \chi \to \delta_j \to \delta_i \cdots (\chi \to \delta_j \to \delta_i) \to (\chi \to \delta_j) \to \chi \to \delta_i, (\chi \to \delta_j) \to \chi \to \delta_i, \chi \to \delta_i, \cdots$$

Die entstandene Folge ist eine C-Ableitung von $\chi \to \varphi$. ⊣

Das Deduktionstheorem ist ein sehr praktisches Werkzeug. Denn Ableitungen im Hilbert-Kalkül (= C-Kalkül) sind recht lang. Mit Hilfe des DTs kommen wir sehr viel schneller zu einem Ergebnis. Zum Beispiel können wir die Transitivität des Schließens ableiten. Dies ist die Formel

$$\texttt{(p0→p1)→(p1→p2)→p0→p2} \tag{21.10}$$

Dazu wenden wir auf (21.10) das DT dreimal an. So bekommen wir (21.11). Es würde anstelle (21.10) nun auch (21.11) genügen.

(21.11) $\texttt{p0}\to\texttt{p1};\texttt{p1}\to\texttt{p2};\texttt{p0}\vdash\texttt{p2}$

Dies abzuleiten, ist keine große Hürde. Hier ist eine Ableitung.

(21.12) $\langle\texttt{p0}\to\texttt{p1},\texttt{p1}\to\texttt{p2},\texttt{p0},\texttt{p1},\texttt{p2}\rangle$

Die ersten Formeln sind unsere Annahmen, und wir wenden MP an. Dies bedeutet nun, dass wir Beweise jetzt etwas kürzer fassen können. Wenn wir eine Formel $\varphi\to\chi$ abgeleitet haben und eine Formel $\chi\to\delta$, dann schreiben wir $(\varphi\to\chi)\to(\chi\to\delta)\to\varphi\to\delta$ hin und wenden zweimal MP an. Diese Formel müsste man nun, wenn man will, an dieser Stelle durch einen Beweis ersetzen. Aber wenn man ihn einmal erbracht hat, so beruft man sich einfach darauf.

Wir benutzen das Deduktionstheorem auch in dem folgenden Beweis.

Satz 21.8 *1. $\chi\to\texttt{f}\vdash\neg\chi$ und $\neg\chi\vdash\chi\to\texttt{f}$.*

2. Genau dann ist $\Delta\vdash\chi$, wenn $\Delta;\neg\chi\vdash\texttt{f}$.

3. Genau dann ist $\Delta;\chi\vdash\texttt{f}$, wenn $\Delta\vdash\neg\chi$.

Beweis. Zu (1). $\langle(\chi\to\texttt{f})\to\neg\chi,\chi\to\texttt{f},\neg\chi\rangle$ ist eine C-Ableitung von $\neg\chi$ aus $\chi\to\texttt{f}$. Die Folge $\langle\neg\chi\to\chi\to\texttt{f},\neg\chi,\chi\to\texttt{f}\rangle$ ist eine C-Ableitung von $\chi\to\texttt{f}$ aus $\neg\chi$. Zu (2). Sei $\Delta\vdash\chi$. Nach (1) ist $\neg\chi\vdash\chi\to\texttt{f}$. Deswegen ist $\Delta;\chi;\neg\chi\vdash\texttt{f}$, und da $\Delta\vdash\chi$, haben wir mit Satz 21.6 (1), $\Delta;\neg\chi\vdash\texttt{f}$. Sei nun $\Delta;\neg\chi\vdash\texttt{f}$. Wir haben auch $\Delta;\chi\to\texttt{f}\vdash\neg\chi$ und so $\Delta;\chi\to\texttt{f}\vdash\texttt{f}$. Da $\texttt{f}\vdash\chi$, so haben wir auch $\Delta;\chi\to\texttt{f}\vdash\chi$. Mit dem Deduktionstheorem haben wir dann $\Delta\vdash\chi\to\texttt{f}\to\chi$. Da $((\chi\to\texttt{f})\to\chi)\to\chi$ Instanz eines Axioms ist, ist $\Delta\vdash\chi$. ⊣

Es heiße eine Menge Δ **konsistent**, falls $\Delta\nvdash\texttt{f}$. Der Satz 21.8 sagt jetzt nichts anderes, als dass φ genau dann aus Δ ableitbar ist, wenn $\Delta;\neg\varphi$ inkonsistent ist.

Wir haben oben gesehen, dass $\varphi\approx\chi$ dasselbe ist wie $\varphi\vDash\chi$ und $\chi\vDash\varphi$. Hier nun haben wir eine ähnliche Tatsache. Es bezeichne $[\chi/p]\delta$ das Resultat der Ersetzung der Variablen p in δ durch χ.

Lemma 21.9 *Ist $\varphi\vdash\chi$ und $\chi\vdash\varphi$ und ist δ eine Formel. Dann ist $[\varphi/p]\delta\vdash[\chi/p]\delta$ sowie $[\chi/p]\delta\vdash[\varphi/p]\delta$.*

Beweis. Der Beweis ist lang aber ohne Schwierigkeit. Zunächst einmal muss p in δ nicht auftreten. Dann ist $[\varphi/p]\delta = [\chi/p]\delta = \delta$. Es genügt für diesen Satz offensichtlich die Tatsache für ein einziges Vorkommen von p zu zeigen. Dies geschieht durch Induktion über die Formel δ. Wir müssen den Fall $\delta = \mathtt{f}$ bzw. $\delta = \mathtt{w}$ nicht betrachten. (Fall 1) $\delta = p$. Dann ist $\delta[\chi/p] = \chi$, $[\varphi/p]\delta = \varphi$, und die Behauptung gilt nach Annahme. (Fall 2) $\delta = q \neq p$. Dann ist $[\chi/p]\delta = [\varphi/p]\delta = q$, und die Behauptung gilt, weil $q \vdash q$. (Fall 2) $\delta = \eta\wedge\theta$. Sei die Behauptung bereits für η und θ gezeigt. Dann haben wir $[\chi/p]\eta \vdash [\varphi/p]\eta$, $[\varphi/p]\eta \vdash [\chi/p]\eta$, $[\chi/p]\theta \vdash [\varphi/p]\theta$, und $[\varphi/p]\theta \vdash [\chi/p]\theta$. Ferner ist $[\chi/p]\delta = ([\chi/p]\eta)\rightarrow([\chi/p]\eta)$. Jetzt bekommen wir

$$
\begin{array}{ll}
 & [\chi/p]\eta\wedge[\chi/p]\theta, \\
 & [\chi/p]\eta\wedge[\chi/p]\theta\rightarrow[\chi/p]\eta, \\
 & [\chi/p]\eta, \\
 & \cdots, \\
 & [\varphi/p]\eta, \\
 & [\chi/p]\eta\wedge[\chi/p]\theta\rightarrow[\chi/p]\theta, \\
(21.13) & [\chi/p]\theta, \\
 & \cdots, \\
 & [\varphi/p]\theta, \\
 & [\varphi/p]\eta\rightarrow[\varphi/p]\theta\rightarrow[\varphi/p]\eta\wedge[\varphi/p]\theta, \\
 & [\varphi/p]\theta\rightarrow[\varphi/p]\eta\wedge[\varphi/p]\theta, \\
 & [\varphi/p]\eta\wedge[\varphi/p]\theta
\end{array}
$$

Dies zeigt $[\chi/p]\eta\wedge[\chi/p]\theta \vdash [\varphi/p]\eta\wedge[\varphi/p]\theta$. (Fall 4) $\delta = \eta\vee\theta$. Ich kürze den Beweis etwas ab:

$$
\begin{array}{ll}
 & [\chi/p]\eta\rightarrow[\varphi/p]\eta, \\
 & [\varphi/p]\eta\rightarrow[\varphi/p]\eta\vee[\varphi/p]\theta, \\
 & \cdots, \\
 & [\chi/p]\eta\rightarrow[\varphi/p]\eta\vee[\varphi/p]\theta, \\
 & [\chi/p]\theta\rightarrow[\varphi/p]\theta, \\
 & [\varphi/p]\theta\rightarrow[\varphi/p]\eta\vee[\varphi/p]\theta, \\
(21.14) & \cdots, \\
 & [\chi/p]\theta\rightarrow[\varphi/p]\eta\vee[\varphi/p]\theta, \\
 & ([\chi/p]\eta\rightarrow[\varphi/p]\eta\vee[\varphi/p]\theta)\rightarrow([\chi/p]\theta\rightarrow[\varphi/p]\eta\vee[\varphi/p]\theta) \\
 & \qquad\rightarrow(([\chi/p]\eta\vee[\chi/p]\theta)\rightarrow([\varphi/p]\eta\vee[\varphi/p]\theta)), \\
 & ([\chi/p]\theta\rightarrow[\varphi/p]\eta\vee[\varphi/p]\theta)\rightarrow(([\chi/p]\eta\vee[\chi/p]\theta)\rightarrow([\varphi/p]\eta\vee[\varphi/p]\theta)), \\
 & ([\chi/p]\eta\vee[\chi/p]\theta)\rightarrow([\varphi/p]\eta\vee[\varphi/p]\theta)
\end{array}
$$

(Fall 5) $\delta = \eta{\to}\theta$.

$$
\begin{array}{ll}
 & [\varphi/p]\eta{\to}[\varphi/p]\theta, \\
 & [\chi/p]\eta{\to}[\varphi/p]\eta, \\
 & ([\varphi/p]\eta{\to}[\varphi/p]\theta){\to}([\chi/p]\eta{\to}[\varphi/p]\eta){\to}[\chi/p]\eta{\to}[\varphi/p]\theta, \\
 & ([\chi/p]\eta{\to}[\varphi/p]\eta){\to}[\chi/p]\eta{\to}[\varphi/p]\theta, \\
(21.15) & [\chi/p]\eta{\to}[\varphi/p]\theta, \\
 & [\varphi/p]\theta{\to}[\chi/p]\theta, \\
 & ([\chi/p]\eta{\to}[\varphi/p]\theta){\to}([\varphi/p]\theta{\to}[\chi/p]\theta){\to}[\chi/p]\eta{\to}[\chi/p]\theta, \\
 & ([\varphi/p]\theta{\to}[\chi/p]\theta){\to}[\chi/p]\eta{\to}[\chi/p]\theta, \\
 & [\chi/p]\eta{\to}[\chi/p]\theta,
\end{array}
$$

(Fall 6) $\delta = \neg\eta$. Dies erledigen wir wie folgt.

$$
\begin{array}{ll}
 & \neg([\chi/p]\eta), \\
 & \neg([\chi/p]\eta){\to}[\chi/p]\eta{\to}\mathtt{f}, \\
 & [\chi/p]\eta{\to}\mathtt{f}, \\
(21.16) & \cdots, \\
 & [\varphi/p]\eta{\to}\mathtt{f}, \\
 & ([\varphi/p]\eta{\to}\mathtt{f}){\to}\neg([\varphi/p]\eta), \\
 & \neg([\varphi/p]\eta)
\end{array}
$$

Damit ist der Beweis abgeschlossen. ⊣

Satz 21.10 (Vollständigkeit) *Genau dann ist $\Sigma \vdash \varphi$, wenn auch $\Sigma \vDash \varphi$.*

Beweis. Gewiss ist $\Sigma \vDash \varphi$, wenn $\Sigma \vdash \varphi$. Dies ist nichts anderes als die Korrektheit der Regeln. Sei also $\Sigma \nvdash \varphi$. Wir zeigen, dass dann auch $\Sigma \nvDash \varphi$. Zunächst einmal haben wir, dass $\Sigma \vdash \varphi$ gleichbedeutend ist mit $\Sigma; \neg\varphi \vdash \mathtt{f}$, sodass $\Sigma \nvdash \varphi$ gleichbedeutend ist mit $\Sigma; \neg\chi \nvdash \mathtt{f}$, dh $\Sigma; \neg\varphi$ ist konsistent. Wir müssen also zeigen, dass eine konsistente Mengen erfüllbar ist. Dies tun wir, indem wir die Erfüllbarkeit schrittweise auf die von einfacheren Mengen zurückführen.

Negation *Genau dann ist $\Sigma; \neg\neg\chi$ konsistent, wenn $\Sigma; \chi$ konsistent ist.* Zum Beweis bemerke ich, dass $\neg\neg\chi \vdash \chi$ und $\chi \vdash \neg\neg\chi$.

Konjunktion *Genau dann ist $\Delta; \varphi\wedge\chi$ konsistent, wenn $\Sigma; \varphi; \chi$ konsistent ist.* Zum Beweis: $\Sigma; \varphi\wedge\chi \vdash \varphi$ sowie $\Sigma; \varphi\wedge\chi \vdash \chi$. Ist also $\Sigma; \varphi; \chi \vdash \mathtt{f}$, so auch $\Sigma; \varphi\wedge\chi \vdash \mathtt{f}$. Die Umkehrung ist ebenso einfach.

Negierte Konjunktion. *Genau dann ist* $\Sigma; \neg(\varphi\wedge\chi)$ *konsistent, wenn* $\Sigma; \neg\varphi$ *oder* $\Sigma, \neg\chi$ *konsistent ist.* Positiv: genau dann ist $\Sigma; \neg(\varphi\wedge\chi) \vdash \mathtt{f}$, wenn $\Sigma; \neg\varphi \vdash \mathtt{f}$ und (!) $\Sigma; \neg\chi \vdash \mathtt{f}$. Sei $\Sigma; \neg(\varphi\wedge\chi) \vdash \mathtt{f}$. Dann $\Sigma \vdash \varphi\wedge\chi$, und so $\Sigma \vdash \varphi$, woraus $\Sigma; \neg\varphi \vdash \mathtt{f}$; und $\Sigma \vdash \chi$, woraus $\Sigma; \neg\chi \vdash \mathtt{f}$. Sei umgekehrt $\Sigma; \neg\varphi \vdash \mathtt{f}$ und $\Sigma; \neg\chi \vdash \mathtt{f}$. Dann $\Sigma \vdash \varphi$ und $\Sigma \vdash \chi$. Daraus folgt $\Sigma \vdash \varphi\wedge\chi$ und so $\Sigma; \neg(\varphi\wedge\chi) \vdash \mathtt{f}$.

Disjunktion. *Genau dann ist* $\Sigma; \varphi\vee\chi$ *konsistent, wenn* $\Sigma; \varphi$ *oder* $\Sigma; \chi$ *konsistent ist.* Oder auch: genau dann ist $\Sigma; \varphi\vee\chi \vdash \mathtt{f}$, wenn $\Sigma; \varphi \vdash \mathtt{f}$ und $\Sigma; \chi \vdash \mathtt{f}$. Es sei also $\Sigma; \varphi\vee\chi \vdash \mathtt{f}$. Sei $\Sigma; \varphi\vee\chi \vdash \mathtt{f}$. Da $\varphi \vdash \varphi\vee\chi$, so erhalten wir $\Sigma; \varphi \vdash \mathtt{f}$. Ebenso sieht man, dass $\Sigma; \chi \vdash \mathtt{f}$. Sei umgekehrt $\Sigma; \varphi \vdash \mathtt{f}$ und $\Sigma; \chi \vdash \mathtt{f}$. Dann ist $\Sigma \vdash \varphi\to\mathtt{f}$ und $\Sigma \vdash \chi\to\mathtt{f}$. Es ist $(\varphi\to\mathtt{f})\to(\chi\to\mathtt{f})\to((\varphi\vee\chi)\to\mathtt{f})$ Instanz von (l$\vee$). Mit dem DT wird dies zu $\varphi\to\mathtt{f}; \chi\to\mathtt{f}; \varphi\vee\chi \vdash \mathtt{f}$. Mit Satz 21.6 bekommen wir $\Sigma; \varphi\vee\chi \vdash \mathtt{f}$.

Negierte Disjunktion. *Genau dann ist* $\Sigma; \neg(\varphi\vee\chi)$ *konsistent, wenn* $\Sigma; \neg\varphi; \neg\chi$ *konsistent ist.* Oder auch: Genau dann ist $\Sigma; \neg(\varphi\vee\chi) \vdash \mathtt{f}$, wenn $\Sigma; \neg\varphi; \neg\chi \vdash \mathtt{f}$. Sei also $\Sigma; \neg(\varphi\vee\chi) \vdash \mathtt{f}$. Dann ist $\Sigma \vdash \varphi\vee\chi$. Nun ist (wie eben festgestellt) $\varphi\to\mathtt{f}; \chi\to\mathtt{f}; \varphi\vee\chi \vdash \mathtt{f}$, also haben wir $\Sigma; \varphi\to\mathtt{f}; \chi\to\mathtt{f} \vdash \mathtt{f}$, oder auch $\Sigma; \neg\varphi; \neg\chi \vdash \mathtt{f}$. Sei umgekehrt $\Sigma; \neg\varphi; \neg\chi \vdash \mathtt{f}$. Da $\varphi \vdash \varphi\vee\chi$, so ist $\neg(\varphi\vee\chi) \vdash \neg\varphi$ sowie $\neg(\varphi\vee\chi) \vdash \neg\chi$. Daraus folgt mit Satz 21.6 (2) $\Sigma; \neg(\varphi\vee\chi) \vdash \mathtt{f}$.

Implikation. *Genau dann ist* $\Sigma; \varphi\to\chi$ *konsistent, wenn* $\Sigma; \neg\varphi$ *oder* $\Sigma; \chi$ *konsistent ist.* Positiv: Genau dann ist $\Sigma; \varphi\to\chi \vdash \mathtt{f}$, wenn $\Sigma; \neg\varphi \vdash \mathtt{f}$ und $\Sigma; \chi \vdash \mathtt{f}$. Sei dazu $\Sigma; \varphi\to\chi \vdash \mathtt{f}$. Dann ist (1) $\Sigma; \chi \vdash \varphi\to\chi \vdash \mathtt{f}$, also $\Sigma; \chi \vdash \mathtt{f}$. Und es ist (2) $\Sigma; \neg\varphi \vdash \varphi\to\chi \vdash \mathtt{f}$, und so $\Sigma; \neg\varphi \vdash \mathtt{f}$. Sei umgekehrt $\Sigma; \neg\varphi \vdash \mathtt{f}$ und $\Sigma; \chi \vdash \mathtt{f}$. Dann gilt $\Sigma \vdash \varphi$ und mit $\Sigma; \varphi; \varphi\to\chi \vdash \chi$ und $\Sigma; \chi \vdash \mathtt{f}$ bekommen wir $\Sigma; \varphi\to\chi \vdash \mathtt{f}$, wie verlangt.

Negierte Implikation. *Genau dann ist* $\Sigma; \neg(\varphi\to\chi)$ *konsistent, wenn* $\Sigma; \varphi; \neg\chi$ *konsistent ist.* Positiv: Genau dann ist $\Sigma; \neg(\varphi\to\chi) \vdash \mathtt{f}$, wenn $\Sigma; \varphi; \neg\chi \vdash \mathtt{f}$. Sei dazu $\Sigma; \neg(\varphi\to\chi) \vdash \mathtt{f}$. Dann ist $\Sigma \vdash \varphi\to\chi$, also $\Sigma; \varphi \vdash \chi$, also $\Sigma; \varphi; \neg\chi \vdash \mathtt{f}$. Umgekehrt folgt aus $\Sigma; \varphi; \neg\chi \vdash \mathtt{f}$ sofort $\Sigma; \varphi \vdash \chi$, weiter $\Sigma \vdash \varphi\to\chi$ und schließlich $\Sigma; \neg(\varphi\to\chi) \vdash \mathtt{f}$.

Sei nun Σ konsistent. Enthalte Σ eine Formel mit einem Operationszeichen verschieden von $\neg$. Aufgrund der obigen Beziehungen finden wir dann eine konsistente Menge Σ', die dieses Zeichen nicht enthält. Ferner wird jede Belegung, die Σ' erfüllt, auch Σ erfüllen. Ich mache dies exemplarisch vor. Es enthalte Σ die Formel $\varphi\to\chi$, es sei also $\Sigma = \Sigma'; \varphi\to\chi$. Da $\Sigma'; \varphi\to\chi$ konsistent ist, ist $\Sigma'; \neg\varphi$ konsistent oder es ist $\Sigma'; \chi$ konsistent. Eine Belegung erfüllt $\Sigma'; \neg\varphi$ oder $\Sigma'; \chi$ genau dann, wenn sie $\Sigma'; \varphi\to\chi (= \Sigma)$ erfüllt. Also ist eine dieser beiden Mengen konsistent, und wir können das Argument mit ihr fortsetzen. Am Schluss erhalten wir eine konsistente Menge, die nur Variable oder negierte Variable enthält. Diese ist aber genau dann konsistent, wenn sie keine Variable zugleich negiert wie nichtnegiert enthält. Denn `p0`; `¬p0` $\vdash$ `f`, da ja `p0` $\vdash$ `p0`, mit Satz 21.8 (2). $\dashv$

Der Beweis ist sehr ähnlich dem Tableau-Kalkül.

Übungen

[76] Leiten Sie folgende Formel ab: `(p0→p1→p2)→p1→p0→p2`.

[77] Zeigen Sie, dass folgende Formel ableitbar ist. `(¬p0→p0)→p0`. (Diese ist verwandt mit der Regel des Clavius, siehe Übung 61.) *Hinweis.* `¬p0` $\vdash$ `p0→f` und `p0→f` $\vdash$ `¬p0`.

[78] Zeigen Sie, dass gilt: Ist $\Delta; \varphi$ konsistent und $\varphi \vdash \chi$, dann ist auch $\Delta; \chi$ konsistent.

Kapitel 22

Modallogik

Bevor wir uns mit der Prädikatenlogik beschäftigen, will ich noch einen neuen Typ von Logik vorstellen, die sogenannte Modallogik. Sie baut auf die bisherige Sprache auf. Wir fügen lediglich dem Alphabet zwei neue Symbole hinzu, $\Box$ und $\Diamond$. Ist φ eine Formel, so sind jetzt auch $\Box\varphi$ und $\Diamond\varphi$ Formeln (wir lassen die äußeren Klammern bereits kommentarlos weg). Man liest $\Box\varphi$ als „notwendig φ" und $\Diamond\varphi$ als „möglich φ", aber dies ist nur eine der vielen Interpretationen, welche die Symbole erhalten können. In der Tat ist die Modallogik ein Paradies für klassische Logiker: während nämlich die Interpretation der klassischen Aussageverbindungen erhalten bleibt, ist die Interpretation der Modaloperatoren frei. Dies erlaubt, sehr viele verschiedene Logiken zu formulieren.

Definition 22.1 (Kripke-Rahmen) *Ein **Kripke-Rahmen** ist ein Paar $\langle W, R\rangle$, wo W eine Menge ist und $R \subseteq W^2$ eine zweistellige Relation auf W. Die Elemente von W heißen **Welten**, und R die **Zugänglichkeitsrelation** auf den Welten.*

Zum Beispiel ist $\langle\{w_0, w_1\}, \{\langle w_0, w_1\rangle\}\rangle$ ein Kripke-Rahmen, in dem es zwei Welten gibt, w_0 und w_1, wobei w_1 von w_0 zugänglich („möglich") ist. Hier ist eine (selbsterklärende) bildliche Darstellung dieses Rahmens.

$$w_0 \bullet \longrightarrow \bullet\, w_1 \tag{22.1}$$

Definition 22.2 (Belegung) *Eine Belegung in einen Kripke-Rahmen $\langle W, R\rangle$ ist eine Funktion β, die jeder Variablen eine Teilmenge von W zuordnet.*

Gegeben eine Belegung β und eine Welt w definieren wir jetzt das Zutreffen einer Formel in w unter der Belegung β.

$$\begin{array}{lll} & \langle\langle W,R\rangle,\beta,w\rangle \vDash p & :\Leftrightarrow w \in \beta(p) \\ & \langle\langle W,R\rangle,\beta,w\rangle \vDash \neg\varphi & :\Leftrightarrow \langle\langle W,R\rangle,\beta,w\rangle \nvDash \varphi \\ & \langle\langle W,R\rangle,\beta,w\rangle \vDash \varphi \wedge \chi & :\Leftrightarrow \langle\langle W,R\rangle,\beta,w\rangle \vDash \varphi;\chi \\ (22.2) & \langle\langle W,R\rangle,\beta,w\rangle \vDash \varphi \vee \chi & :\Leftrightarrow \langle\langle W,R\rangle,\beta,w\rangle \vDash \varphi \text{ oder } \langle\langle W,R\rangle,\beta,w\rangle \vDash \chi \\ & \langle\langle W,R\rangle,\beta,w\rangle \vDash \varphi \to \chi & :\Leftrightarrow \langle\langle W,R\rangle,\beta,w\rangle \nvDash \varphi \text{ oder } \langle\langle W,R\rangle,\beta,w\rangle \vDash \chi \\ & \langle\langle W,R\rangle,\beta,w\rangle \vDash \Box\varphi & :\Leftrightarrow \text{für alle } w' \text{ mit } w\,R\,w'\text{: } \langle\langle W,R\rangle,\beta,w'\rangle \vDash \varphi \\ & \langle\langle W,R\rangle,\beta,w\rangle \vDash \Diamond\varphi & :\Leftrightarrow \text{es gibt } w' \text{ mit } w\,R\,w' \text{ und} \\ & & \qquad\qquad \langle\langle W,R\rangle,\beta,w'\rangle \vDash \varphi \end{array}$$

Ich gebe ein etwas größeres Beispiel. Wir wählen uns $\langle\mathbb{N},<\rangle$ zum Kripke-Rahmen. Ferner sei $\beta(p_0)$ die Menge der Primzahlen, $\beta(p_1)$ die Menge der geraden Zahlen und $\beta(p_2) = \{0,1\}$. Dann gilt zum Beispiel:

$$(22.3) \qquad \langle\langle\mathbb{N},<\rangle,\beta,0\rangle \vDash \Box(p_0 \to \Diamond p_0)$$

Denn dazu muss für alle $i > 0$ gelten: es ist $\langle\langle\mathbb{N},<\rangle,\beta,i\rangle \nvDash p_0$ (also i keine Primzahl) oder aber

$$(22.4) \qquad \langle\langle\mathbb{N},<\rangle,\beta,i\rangle \vDash \Diamond p_0$$

Dies bedeutet nichts anderes, als dass es ein $j > i$ gibt, für das $\langle\langle\mathbb{N},<\rangle,\beta,j\rangle \vDash p_0$, dh bei dem j eine Primzahl ist. (22.3) gilt, weil es unendlich viele Primzahlen gibt! Ebenso gilt

$$(22.5) \qquad \langle\langle\mathbb{N},<\rangle,\beta,0\rangle \vDash \Box(p_1 \to \Diamond p_1)$$

weil es unendlich viele gerade Zahlen gibt. Hingegen haben wir

$$(22.6) \qquad \langle\langle\mathbb{N},<\rangle,\beta,0\rangle \vDash \Diamond(p_2 \wedge \neg\Diamond p_2)$$

Denn es ist

$$(22.7) \qquad \langle\langle\mathbb{N},<\rangle,\beta,1\rangle \vDash p_2; \neg\Diamond p_2$$

da ja p_2 in keiner Welt $j > 1$ wahr ist.

Es gilt nun ebenfalls

$$(22.8) \qquad \langle\langle\mathbb{N},<\rangle,\beta,0\rangle \vDash \Diamond p_0 \Diamond p_1 \to \Diamond(p_0 \wedge p_1) \vee \Diamond(p_0 \wedge \Diamond p_1) \vee \Diamond(p_1 \wedge \Diamond p_0)$$

Dies kann man nun einerseits für die gegebene Belegung direkt nachrechnen. Ich behaupte aber, dass dies *für jede Belegung und in jeder Welt gilt*. Sei dazu i eine Welt, γ eine Belegung. Und es sei

$$\langle\langle\mathbb{N}, <\rangle, \gamma, i\rangle \vDash \Diamond p_0 \Diamond p_1 \tag{22.9}$$

Dann existieren $j, k > i$ mit

$$\langle\langle\mathbb{N}, <\rangle, \gamma, j\rangle \vDash p_0, \quad \langle\langle\mathbb{N}, <\rangle, \gamma, k\rangle \vDash p_1 \tag{22.10}$$

Fall 1. $j = k$. Dann ist

$$\langle\langle\mathbb{N}, <\rangle, \gamma, j\rangle \vDash p_0 \wedge p_1 \tag{22.11}$$

und somit

$$\langle\langle\mathbb{N}, <\rangle, \gamma, i\rangle \vDash \Diamond(p_0 \wedge p_1) \tag{22.12}$$

Fall 2. $j < k$. Dann ist

$$\langle\langle\mathbb{N}, <\rangle, \gamma, j\rangle \vDash p_0; \Diamond p_1 \tag{22.13}$$

und somit

$$\langle\langle\mathbb{N}, <\rangle, \gamma, i\rangle \vDash \Diamond(p_0 \wedge \Diamond p_1) \tag{22.14}$$

Fall 3. $j > k$. Dann ist

$$\langle\langle\mathbb{N}, <\rangle, \gamma, k\rangle \vDash p_1; \Diamond p_0 \tag{22.15}$$

und somit

$$\langle\langle\mathbb{N}, <\rangle, \gamma, i\rangle \vDash \Diamond(p_1 \wedge \Diamond p_0) \tag{22.16}$$

Ist $\langle W, R\rangle$ ein Kripke-Rahmen, so bilden die Formeln, welche in $\langle W, R\rangle$ in jeder Welt unter jeder Belegung wahr sind, eine sogenannte normale Modallogik.

Definition 22.3 (Modallogik) *Eine* ***normale Modallogik*** *ist eine Formelmenge L, die folgende Eigenschaften hat:*

1. *Ist $\varphi \in Auss$ eine Tautologie, so ist $\varphi \in L$.*
2. $\Box(p_0 \to p_1) \to \Box p_0 \to \Box p_1 \in L$.

3. $\Box p_0 \to \neg\Diamond\neg p_0, \neg\Diamond\neg p_0 \to \Box p_0 \in L$.

4. *Ist* $\varphi \in L$*, so auch* $h_\sigma(\varphi) \in L$*,* σ *eine beliebige Ersetzung.*

5. *Ist* $\varphi \to \chi \in L$ *und* $\varphi \in L$*, so auch* $\chi \in L$*.*

6. *Ist* $\varphi \in L$*, so auch* $\Box\varphi \in L$*.*

Die kleinste normale Modallogik wird mit K *bezeichnet.*

Man beachte, dass die Sprache *Auss* wie in Abschnitt 17 definiert ist. Das bedeutet, dass genau dann $\varphi \in \mathit{Auss}$ ist, wenn φ die Operatoren $\Box$ und $\Diamond$ gar nicht enthält. In diesem Fall ist φ ein Tautologie, wenn es wie in 17 beschrieben unter allen Belegungen wahr ist.

Da die Logik unter Substitution abgeschlossen ist, bekommen wir auch, dass $\Box p_0 \vee \neg\Box p_0 \in L$ ist, obwohl dies nicht in *Auss* ist. Jedoch ist $p_1 \vee \neg p_1 \in \mathit{Auss}$ und die Ersetzung $p_1 \mapsto \Box p_0$ überführt dies in die erste Formel. Also ist diese in L für jedes L.

Satz 22.4 *Sei* $\langle W, R\rangle$ *ein Kripke-Rahmen. Dann ist* $\mathsf{Th}\,\langle W, R\rangle$ *eine normale Modallogik.*

$$\mathsf{Th}\,\langle W, R\rangle := \{\varphi : \textit{für alle } \beta, w\text{: } \langle\langle W, R\rangle, \beta, w\rangle \vDash \varphi\} \tag{22.17}$$

Beweis. Dazu muss man zeigen, dass diese Menge die Bedingungen erfüllt. So müssen wir zeigen, dass (1) für eine klassische Tautologie φ gilt $\langle\langle W, R\rangle, \beta, w\rangle \vDash \varphi$. Dazu setze $\gamma(p_i) := 1$ falls $w \in \beta(p_i)$ und $\gamma(p_i) = 0$ sonst. Man zeigt dann leicht induktiv, dass für eine klassische Tautologie φ gilt $\langle\langle W, R\rangle, \beta, w\rangle \vDash \varphi$ gdw. $\overline{\gamma}(\varphi) = 1$. Ist also φ eine klassische Tautologie, so ist $\langle\langle W, R\rangle, \beta, w\rangle \vDash \varphi$ für jedes β und jedes w. (2) Weiterhin sei $\langle\langle W, R\rangle, \beta, w\rangle \vDash \Box(p_0 \to p_1); \Box p_0$. Sei w' derart, dass $w\,R\,w'$. Dann ist $\langle\langle W, R\rangle, \beta, w'\rangle \vDash p_0 \to p_1; p_0$, und deswegen $\langle\langle W, R\rangle, \beta, w'\rangle \vDash p_1$. Da w' beliebig war, ist $\langle\langle W, R\rangle, \beta, w\rangle \vDash \Box p_0$. (3) Sei $\langle\langle W, R\rangle, \beta, w\rangle \vDash \Box p_0$. Zu zeigen ist $\langle\langle W, R\rangle, \beta, w\rangle \nvDash \Diamond\neg p_0$. Denn sei $\langle\langle W, R\rangle, \beta, w\rangle \vDash \Diamond\neg p_0$. Dann existiert ein w' mit $w\,R\,w'$ und $\langle\langle W, R\rangle, \beta, w'\rangle \vDash \neg p_0$, was nach Voraussetzung nicht gilt. Umgekehrt folgt aus $\langle\langle W, R\rangle, \beta, w\rangle \vDash \neg\Diamond\neg p_0$ sofort $\langle\langle W, R\rangle, \beta, w\rangle \vDash \Box p_0$. (4) Es sei $\varphi \in \mathsf{Th}\,\langle W, R\rangle$. Sei β eine Belegung in $\langle W, R\rangle$. Dann definiere $\gamma(p) := \{w : \langle\langle W, R\rangle, \beta, w\rangle \vDash h_\sigma(p)\}$. Man beweist dann durch Induktion, dass $\langle\langle W, R\rangle, \gamma, w\rangle \vDash \varphi$ genau dann, wenn $\langle\langle W, R\rangle, \beta, w\rangle \vDash h_\sigma(\varphi)$. Dann nun φ auch unter γ in w wahr ist, ist $h_\sigma(\varphi)$ unter β in w wahr. β und w waren beliebig gewählt. Also ist $h_\sigma(\varphi) \in \mathsf{Th}\,\langle W, R\rangle$. (5) Es sei $\varphi \to \chi \in \mathsf{Th}\,\langle W, R\rangle$ und $\chi \in \mathsf{Th}\,\langle W, R\rangle$. Sei β eine Belegung und w eine Welt. Dann ist $\langle\langle W, R\rangle, \beta, w\rangle \vDash \varphi \to \chi; \varphi$. Also ist $\langle\langle W, R\rangle, \beta, w\rangle \vDash \chi$. (6) Es sei $\varphi \in \mathsf{Th}\,\langle W, R\rangle$. Dann ist auch $\Box\varphi \in \mathsf{Th}\,\langle W, R\rangle$. Denn sei β eine Belegung, $w \in W$. Dann gilt nach Voraussetzung für jedes w' $\langle\langle W, R\rangle, \beta, w'\rangle \vDash \varphi$. Also gilt für jedes w' mit $w\,R\,w'$, dass $\langle\langle W, R\rangle, \beta, w'\rangle \vDash \varphi$. Also ist $\langle\langle W, R\rangle, \beta, w\rangle \vDash \Box\varphi$. β und w waren beliebig. Also ist $\Box\varphi \in \mathsf{Th}\,\langle W, R\rangle$. $\dashv$

Sind L und L' normale Modallogiken, so auch $L \cap L'$. Das ist nicht schwer zu zeigen. Ferner ist mit L_i, $i \in I$, auch der Schnitt $\bigcap_{i \in I} L_i$ eine normale Modallogik. Daraus folgt

Satz 22.5 *Es sei $\mathcal{K}$ eine beliebige Klasse von Kripke-Rahmen. Dann ist* $\mathsf{Th}\,\mathcal{K}$ *ebenfalls eine Logik, wo*

$$\mathsf{Th}\,\mathcal{K} := \bigcap_{\langle W,R\rangle \in \mathcal{K}} \mathsf{Th}\,\langle W, R\rangle \qquad (22.18)$$

$\dashv$

Ich schließe mit dem Beispiel einer wichtigen Logik. Es sei $\mathcal{K}_u$ die Klasse aller Kripke-Rahmen $\langle W, R\rangle$, wo $R = W^2$ ist. Wir setzen

$$\mathsf{S5} := \mathsf{Th}\,\mathcal{K}_u \qquad (22.19)$$

Satz 22.6 $\mathsf{S5}$ *ist die kleinste Modallogik, die die Formeln $\Box p_0 \to p_0$, $p_0 \to \Box\Diamond p_0$ und $\Box p_0 \to \Box\Box p_0$ enthält.*

Lesen wir „$\Box$” wie oben vorgeschlagen als „notwendig” und „$\Diamond$” als „möglich”, so sind die Postulate von $\mathsf{S5}$ durchaus motiviert. Zum Beispiel: falls eine Aussage notwendig ist, so ist sie auch wahr. Oder: falls etwas wahr ist, so ist es notwendig, dass diese Aussage möglicherweise wahr ist. Und wenn eine Aussage notwendigerweise wahr ist, so ist es notwendig, dass sie notwendigerweise wahr ist. Semantisch gesehen besteht $\mathsf{S5}$ aus Strukturen, in denen jede Welt von jeder Welt aus zugänglich ist.

Jedoch gibt es Kripke-Rahmen, deren Logik $\mathsf{S5}$ ist, ohne dass sie aus $\mathcal{K}_u$ sind. Ein solcher Kripke-Rahmen ist $\langle\{w_0, w_1\}, \{\langle w_0, w_0\rangle, \langle w_1, w_1\rangle\}\rangle$.

Übungen

[79] Es sei $\langle\mathbb{N}, <\rangle$ gegeben und β wie oben definiert. Bestimmen Sie, ob gilt:

$$\langle\langle\mathbb{N}, <\rangle, \beta, 0\rangle \vDash \Box((\neg p_1) \to \Diamond\Diamond p_0) \qquad (22.20)$$

[80] Es sei $F_2 := \langle\{w_0, w_1\}, \{\langle w_0, w_0\rangle, \langle w_0, w_1\rangle, \langle w_1, w_0\rangle, \langle w_1, w_1\rangle\}\rangle$. Zeigen Sie:

1. F_2 ist in $\mathcal{K}_u$.
2. $\Box p_0 \to p_0 \in \mathsf{Th}\, F_2$.
3. $p_1 \to \Diamond p_1 \in \mathsf{Th}\, F_2$.

[81] Es sei $G = \langle W, R\rangle$ in $\mathcal{K}_u$, und β eine Belegung. Überlegen Sie, dass entweder für *alle* $w \in W$ gilt $\langle\langle W, R\rangle, \beta, w\rangle \vDash \Diamond p_0$ oder für alle $w \in W$ $\langle\langle W, R\rangle, \beta, w\rangle \nvDash \Diamond p_0$.

[82] Zeigen Sie, dass $\langle\{w_0, w_1\}, \{\langle w_0, w_0\rangle, \langle w_1, w_1\rangle\}\rangle$ die Axiome von $\mathsf{S5}$ erfüllt. (Anmerkung. Sie müssen offenkundig nur zeigen, dass die drei zusätzlichen Formeln in allen Welten gelten. Dafür genügt es zu zeigen, dass sie in w_0 unter jeder Belegung erfüllt sind.)

Kapitel 23

Prädikatenlogik

Die Prädikatenlogik ist in gewissem Maße eine Erweiterung der Aussagenlogik. In der Prädikatenlogik sind Aussagen aber nicht mehr primitiv, sondern sie sind syntaktisch zusammengesetzt. Der einfachste Typ einer Aussage ist die Behauptung, dass ein Gegenstand unter ein Prädikat fällt, etwa der Satz /`Karl ist groß.`/. Er sagt, dass ein Individuum, hier benannt durch den Namen „Karl", eine Eigenschaft hat, nämlich groß zu sein. Eigenschaften und Relationen werden in der Prädikatenlogik einheitlich als **Prädikate** aufgefasst. Prädikate sind Ausdrücke, die aus Termen Aussagen machen. Was Terme sind, wird noch zu klären sein; Eigennamen gehören auf jeden Fall dazu. Die Prädikatenlogik ist sehr stark in ihrer Ausdruckskraft, da sie außer Behauptungen über Individuen auch allgemeine Aussagen mit Hilfe von Quantoren formulieren kann; sie kann sogar Funktionen ausdrücken. (Diese sind semantisch auf Prädikate zurückführbar, aber man tut dies in der Praxis nicht, weil das nur verwirrt.)

Die Syntax der Prädikatenlogik ist anders als die der natürlichen Sprache. Wie in der Aussagenlogik ist man um Eindeutigkeit bemüht. Es darf nicht vorkommen, dass man einen Satz so oder so lesen kann. In der natürlichen Sprache etwa kann man Sätze sagen wie

(23.1) `Ich mag große Wälder und Städte.`

Aber es ist nicht klar, ob ich damit gesagt habe, dass ich Städte mag oder dass ich große Städte mag. Um also Missverständnisse zu verhindern, muss die Syntax anders aussehen.

In der Prädikatenlogik symbolisiert man den Satz /`Karl ist groß.`/ durch

(23.2) `groß`(`Karl`)

Dabei ist die Zeichenkette /`groß`/ jetzt sowohl eine Kette des Deutschen wie der Prädikatenlogik. Natürlich soll dies suggerieren, dass sie dasselbe bedeuten, auch wenn das formale Symbol eine andere Gebrauchsweise besitzt. In der Semantik schreibt man deswegen oft wie folgt.

(23.3) $\mathsf{groß}'(\mathsf{karl}')$

Dabei soll der Wechsel im Font (sans-serife) sowie das Apostroph signalisieren, dass es sich um Elemente einer formalen Sprache handelt. Ich will jedoch an dieser Stelle diesen Unterschied nicht machen, weil das primäre Studienobjekt gar nicht die natürlichen Sprachen sind sondern die formalen.

Das Prädikat ist hier das Symbol /`groß`/, weil es eine Eigenschaft bezeichnet. Das Prädikat wird vor ein Klammerpaar geschrieben, welches die Argumente enthält. Das einzige Argument ist hier „Karl". Ähnlich symbolisiert man die Aussage /`Peter ist ein Nachbar von Karl.`/ durch

(23.4) `nachbar`(`peter`, `karl`)

Wiederum kommt das Prädikat zuerst, und dann kommen die Argumente. Es kommt dabei auf die Reihenfolge an; /`Peter ist größer als Karl.`/ bedeutet nicht dasselbe wie /`Karl ist größer als Peter.`/. Die folgenden Formeln sind also erst einmal verschieden, und zwar sowohl syntaktisch wie auch (in diesem Fall) semantisch.

(23.5) `größer(peter,karl)`

(23.6) `größer(karl,peter)`

Manchmal kann natürlich die Bedeutung identisch sein, wie etwa in

(23.7) `nachbar(peter,karl)`

(23.8) `nachbar(karl,peter)`

Anders als Nachbarschaft ist die Relation „größer als" nicht symmetrisch.

Es gibt nun Prädikate unterschiedlicher Komplexität. „klug" verlangt nur ein Argument, „ist Nachbar von" zwei, „ist am $\cdots$ in $\cdots$ gewesen" verlangt drei (eine Person oder eine Sache, einen Zeitpunkt und einen Ort). Wenn wir nun „klug" durch ein Prädikat /`klug`/, „Nachbar" durch ein Prädikat /`nachbar`/ und „ist am$\cdots$in$\cdots$gewesen" durch ein Prädikat /`gewesen`/ wiedergeben, so sind die folgenden Formeln syntaktisch wohlgeformt:

(23.9) `klug(peter)`

(23.10) `nachbar(karl,peter)`

(23.11) `gewesen(karl,berlin,montag)`

Die folgenden Formeln sind hingegen nicht wohlgeformt:

(23.12) `klug(peter,berlin)`

(23.13) `nachbar(karl)`

(23.14) `gewesen(karl,hamburg)`

Offenkundig kommt es auf die Anzahl der Argumente an. Die Prädikate haben immer dieselbe Anzahl Argumente und dürfen nur genau mit so vielen Argumenten verwendet werden.

Ein weiterer Unterschied, den man machen muss, ist der zwischen **Relationen** und **Funktionen**. Syntaktisch gesehen sind Funktionen Ausdrücke, die aus Termen wieder Terme formen. Eine Funktion benötigt Argumente und liefert dann einen Wert. So ist etwa die Addition (+) eine Funktion. Gegeben zwei Zahlen liefert sie eine Zahl. Genauso funktionieren Ausdrücke wie „der Präsident", „der Vorstandsvorsitzende", „die Muttersprache". Zu einem Objekt gibt es einen Besitzer, zu einer Firma einen Vorstandsvorsitzenden, und jede Person hat eine Muttersprache. (Nur so nebenbei sei erwähnt, dass die meisten Ausdrücke nicht wirklich Funktionen sondern eher partielle Funktionen bezeichnen, aber auf diese Feinheiten will ich nicht weiter eingehen.) Die Syntax von Funktionszeichen ist wie die der Prädikate. Wir schreiben also wie folgt.

(23.15) `präsident(frankreich)`
der Präsident von Frankreich

(23.16) `vorstandsvorsitzende(daimler)`
der Vorstandsvorsitzende von Daimler

(23.17) `muttersprache(karl)`
die Muttersprache von Karl

Hierbei sind /`frankreich`/, /`daimler`/ und /`karl`/ sogenannte **Konstanten**, landläufig auch einfach **Namen** genannt. Diese bezeichnen ein bestimmtes Objekt (oder eine bestimmte Person). Anstelle von Namen können wir auch Variable verwenden. Im Unterschied zu Namen ist ihre Bedeutung nicht vorher festgelegt. Ihre

Funktion wird insbesondere bei Quantoren noch deutlich werden. Ich erlaube nun auch im „Deutschen” (genauer in unserer formalen Version davon) den Gebrauch von Variablen. So können wir die Formeln wie folgt paraphrasieren.

(23.18) `präsident(x0)`
der Präsident von x0

(23.19) `vorstandsvorsitzende(x12)`
der Vorstandsvorsitzende von x12

(23.20) `muttersprache(x6)`
die Muttersprache von x6

Die eben gezeigten Formeln sind allesamt, wie man sagt, **atomar**. Sie sind lediglich aus Namen, Variable, Funktionen und Prädikaten gebildet worden. Dabei darf man so viele Symbole verwenden wir man möchte. Aus syntaktischen Gründen aber kann eine atomare Formel nur ein einziges Relationszeichen enthalten, jedoch im Prinzip beliebig viele Konstanten und Funktionszeichen. So sind die folgenden Formeln atomar:

(23.21) `7+x0>(x0+1)*(x0-1)`
x0 plus 7 ist größer als das Produkt aus x0 vermehrt um 1 und x0 vermindert um 1

(23.22) `größer(direktor(notenbank(montenegro)),karl)`
der Direktor der Notenbank von Montenegro ist größer als Karl

Wir können nun als Erstes die aussagenlogischen Junktoren verwenden und damit komplexe Aussagen bilden.

(23.23) `¬gewesen(karl,berlin,montag)`

(23.24) `nachbar(karl,peter)→größer(karl,alex)`

Diese können wir wir folgt wiedergeben.

(23.25) Karl ist nicht am Montag in Berlin gewesen.

(23.26) Ist Karl Nachbar von Peter, so ist Karl größer als Alex.

Zusätzlich dazu hat die Prädikatenlogik auch noch sogenannte **Quantoren**. Diese sind „∃” (lies: *es gibt* oder *es existiert*) sowie „∀” (lies: *für alle* oder jeder/jede/jedes).

Die Quantoren sind allerdings etwas anders als unsere natürlichen, weil sie erstens immer mit einer Variablen auftreten müssen, also etwa „`∃x7`" (lies: *es existiert ein x7*) „`∀x15`" (lies: *(für) alle x15*). Zweitens aber erlauben sie zunächst einmal nur, ein einziges Prädikat zu verwenden. Syntaktisch gesehen sind sie im Verbund mit ihrer Variablen einstellige Satzoperatoren.

(23.27) `(∃x7)größer(x7,karl)`
Es gibt ein x7, sodass x7 größer ist als Karl.

(23.28) `(∀x3)nachbar(x3,karl)`
Für alle x3, x3 ist Nachbar von Karl.

Am Schluss werde ich noch darauf zurückkommen, wie man sich etwas weniger gestelzt ausdrücken kann. Doch für den Moment ist es wichtig, dass man versteht, wie die Formeln zu lesen sind.

Die Prädikatenlogik ist nun gar nicht eine einzige Sprache sondern ein ganzes Bündel. Wir dürfen nämlich die Grundrelationen und Grundfunktionen und Konstanten frei wählen. In der Regel ist nur eine Grundrelation immer dabei: die Gleichheit, =. Da Prädikate verschieden viele Argumente haben können, muss man also, bevor die Sprache feststeht, zunächst einmal sagen, welche Grundrelationen man annimmt und wieviel Argumente diese jeweils verlangen. Diese Angabe nennt man eine **Signatur**.

Zusätzlich zu Prädikaten haben wir auch Funktionen. Auch wenn technisch verzichtbar, ist der Notationsaufwand ohne Funktionen sehr viel höher. In der Prädikatenlogik dürfen wie beliebig viele Funktionen mit beliebiger Stellenzahl haben.

Definition 23.1 (Signatur) *Eine **Signatur** ist ein Paar $\langle F, \Omega\rangle$, wo F eine Menge und $\Omega : F \to \mathbb{N}$. Wir schreiben auch gerne Ω für diese Signatur. Es heißt $f \in F$ **einstellig** (**zweistellig**, **dreistellig**), falls $\Omega(f) = 1$ ($\Omega(f) = 2$, $\Omega(f) = 3$).*

Definition 23.2 (Syntax) *Es seien $\langle F, \Omega\rangle$ und $\langle G, \Psi\rangle$ zwei Signaturen. Dann sei*

(23.29) $A_{\Omega;\Psi} := F \cup G \cup \{\mathtt{w}, \mathtt{f}, \mathtt{(}, \mathtt{)}, \wedge, \vee, \neg, \rightarrow, \exists, \forall, =, \mathtt{,}, \mathtt{x}, \mathtt{0}, \cdots, \mathtt{9}\}$

*das Alphabet der zugehörigen prädikatenlogischen Sprache. (Wir setzen voraus, dass die drei Mengen paarweise disjunkt sind.) In diesem Zusammenhang ist Ω die **Funktionalsignatur** und Ψ die **Relationalsignatur**. $\vec{x} \in P^*$ ist eine **Variable**, falls $\vec{x}$ die Form $\mathtt{x}\vec{u}$ hat, wo $\vec{u}$ eine Ziffernfolge ist. (Die Konvention ist die gleiche, wie bei den Aussagen, siehe Kapitel 17.) Ein **Term** ist*

① *eine Variable, oder*

② *ein Ausdruck der Form* $f\,(\vec{u}_0\,\texttt{,}\,\vec{u}_1\,\texttt{,}\,\cdots\,\texttt{,}\,\vec{u}_{n-1}\texttt{)}$*, wo* $f \in F$*,* $n = \Omega(f)$ *und wo die* $\vec{u}_i$*,* $i < n$*, Terme sind.*

Eine Zeichenkette heißt eine ***Formel****, wenn eines der Folgenden gilt:*

① $\vec{x} = P\,\texttt{(}\vec{v}_0\,\texttt{,}\,\vec{v}_1\,\texttt{,}\cdots\texttt{,}\,\vec{v}_{n-1}\texttt{)}$*, wo* $P \in G$*,* $n = \Psi(P)$*, und wo die* $\vec{v}_i$*,* $i < n$*, Terme sind. (Diese heißen auch* ***atomare*** *Formeln.)*

② $\vec{x} = \texttt{(}\vec{y}\texttt{=}\vec{z}\texttt{)}$*, wo* $\vec{y}$ *und* $\vec{z}$ *Terme sind.*

③ $\vec{x} = \texttt{w}$ *oder* $\vec{x} = \texttt{f}$*,*

④ $\vec{x} = \texttt{(}\neg\vec{y}\texttt{)}$ *für eine Aussage* $\vec{y}$*.*

⑤ $\vec{x} = \texttt{(}\vec{y}\wedge\vec{z}\texttt{)}$ *für Aussagen* $\vec{y}$ *und* $\vec{z}$*.*

⑥ $\vec{x} = \texttt{(}\vec{y}\vee\vec{z}\texttt{)}$ *für Aussagen* $\vec{y}$ *und* $\vec{z}$*,*

⑦ $\vec{x} = \texttt{(}\vec{y}\rightarrow\vec{z}\texttt{)}$ *für Aussagen* $\vec{y}$ *und* $\vec{z}$*.*

⑧ $\vec{x} = \texttt{(}\forall\vec{v}\texttt{)}\vec{y}$*, wo* $\vec{v}$ *eine Variable ist und* $\vec{y}$ *eine Formel.*

⑨ $\vec{x} = \texttt{(}\exists\vec{v}\texttt{)}\vec{y}$*, wo* $\vec{v}$ *eine Variable ist und* $\vec{y}$ *eine Formel.*

Sei zum Beispiel $F := \{\texttt{+}, \texttt{*}, \texttt{-}\}$, und $\Omega(\texttt{+}) := \Omega(\texttt{*}) := 2$, sowie $\Omega(\texttt{-}) = 1$. Dann sind die folgenden Dinge Terme:

(23.30) `x0`, `*(x1,+(x0,-(x1)))`, `*(+(x0,x1),x0)`, `*(x1,x1)`

Dies können wir sehr kompakt in einer kontextfreien Grammatik hinschreiben.

(23.31)

```
<Formel>  →  (<Formel>→<Formel>) | (<Formel>∧<Formel>)
             | (<Formel>∨<Formel>) | (¬<Formel>)
             | (∀<Var>)<Formel> | (∃<Var>)<Formel>
             | w | f | <1-Rel>(<Term>)
             | <2-Rel>(<Term>,<Term>) | (<Term>=<Term>)
             | <3-Rel>(<Term>,<Term>,<Term>) | ···
<Term>    →  <0-Fkt>() | <Var> | <1-Fkt>(<Term>)
             | <2-Fkt>(<Term>,<Term>) | ···
<Var>     →  x<Index>
<Index>   →  <Ziffer> | <Index><Ziffer>
<Ziffer>  →  0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
```

Dazu kämen dann noch für jedes n-stellige Relationssymbol R (dh für jedes R mit $\Psi(R) = n$) die Regel

(23.32) $\langle n\text{-}\mathtt{Rel}\rangle \to R$

sowie für jedes n-stellige Funktionssymbol f (dh für jedes f mit $\Omega(f) = n$) die Regel

(23.33) $\langle n\text{-}\mathtt{Fkt}\rangle \to f$

Wie schon in der Aussagenlogik werden wir diese recht inhumane Syntax auflockern. Zweistellige Funktionszeichen werden wir zwischen die Argumente stellen, und äußere Klammern werden entfallen. Relationsszeichen binden stets stärker als die aussagenlogischen Junktoren, also ¬, ∧, ∨ und →. Damit wird aus den in (23.30) gezeigten Zeichenketten

(23.34) `x0,x1*(x0+(-(x1)),(x0+x1)*x0,x1*x1`

Bei einstelligen Funktionszeichen sparen wir ebenfalls die Klammern. Man beachte, dass in dieser Signatur das Minuszeichen als einstellig vorgegeben wurde. Das entspricht nicht unserem normalem Gebrauch, da wir zum Beispiel /`7-5`/ anstelle von /`7+(-5)`/ schreiben. Der alltägliche Gebrauch ist jedoch ein Problemfall, weil es auch das einstellige Symbol gibt. Denn `-7` ist eine Zahl, und so ist `-` sowohl einstellig als auch zweistellig. Dafür ist hier allerdings kein Platz. Man muss sich für eine der Stelligkeiten entscheiden. Eine Möglichkeit, das zweistellige Minuszeichen zu eliminieren, ist, es durch Addition mit der Minuszahl zu ersetzen also etwa `7-5` durch `7+(-5)`.

Es gibt noch einen Sonderfall zu besprechen. Tritt der Fall $\Omega(f) = 0$ ein, so müssten wir theoretisch f`()` schreiben. Nullstellige Funktionen, wie diese f auch heißen, werden normalerweise als **Konstanten** bezeichnet. Wir schreiben dafür aber schlicht f. Namen sind so ein Fall. Der Name /`Karl`/, sofern er eine eindeutig bestimmte Person bezeichnet, ist eine Konstante. Die Zahl /`1`/ ist auch eine Konstante. So schreiben wir jetzt

(23.35) `x0*x0+1`

anstelle des korrekten

(23.36) `+(*(x0,x0),1())`

Dabei wurde auch noch Addition als schwächer bindend als Multiplikation benutzt.

Ebenso werden wir bei 2-stelligen Relationszeichen das Relationssymbol zwischen seine Argumente setzen, wie wir dies schon bei der Gleichheit getan haben. In diesem Fall sind die Klammern optional. Wir schreiben also wahlweise $/\vec{x}=\vec{y}/$ oder $/(\vec{x}=\vec{y})/$. Ansonsten ist die Notation so wie oben angegeben: also erst das Prädikatsymbol, dann eine öffnende Klammer, dann die Argumente, getrennt durch Komma, und ein schließende Klammer.

Ich werde im Folgenden **Ausdruck** sagen, wenn ich eine im Sinne der Syntax wohlgeformte Zeichenkette meine, aber den Gebrauch von Konventionen erlaube. Es sei nun $\langle\{<\}, \Psi\rangle$ mit $\Psi(<) := 2$ eine relationale Signatur, und $\langle\{+, *, -, 0, 1\}, \Omega\rangle$ wie oben. Dann sind die folgenden Ausdrücke Formeln:

(23.37) `0<1`, `x0=x1→x0+x2=x1+x2`

Mit Klammern sähen sie zum Beispiel so aus:

(23.38) `(0<1)`, `(x0=x1)→(x0+x2=x1+x2)`

(Wiederum kann man in der zweiten Formeln noch einmal Klammern setzen, aber wie gesagt, dies sind alles optionale Varianten.) Diese sind auch sämtlich wahr. Schauen wir uns diesen Ausdruck an.

(23.39) `(¬(x2=1)∧∀x0)(∃x1)((x0*x1=x2)→(x0=1∨x0=x2))`

Dies ist eine Formel. Sie sagt Folgendes über eine Zahl p (repräsentiert durch die Variable `x2`): diese Zahl ist nicht 1; und für jede Zahl m, sofern es eine Zahl n gibt mit $mn = p$, so ist $n = 1$ oder $n = p$. Hier ist m repräsentiert durch `x0` und n durch `x1`. In den natürlichen Zahlen bedeutet dies nichts anderes, als dass p Primzahl ist. Also ist (23.39) die prädikatenlogische Übersetzung von *`x2 ist Primzahl`*.

Wie kann man nun Ausdrücke der natürlichen Sprache in Prädikatenlogik umsetzen? Dazu ein paar informelle Beispiele. Ein intransitiver Satz wie /`Karl studiert.`/ mit einem Eigennamen als Subjekt wird formalisiert, indem man für den Eigennamen eine Konstante (oder 0-stellige Funktion) `karl` annimmt und für das Verb eine 1-stellige Relation `studiert`. Dann schreibt sich das Ergebnis

(23.40) `studiert(karl)`

Gewöhnliche Nomina wie /`Mensch`/, /`Katze`/, /`Tisch`/ sind gleichfalls (in der Regel) 1-stellige Relationen, ebenso Adjektive /`rot`/, /`billig`/ oder /`schrottreif`/.

Interessant sind intransitive Sätze, die mit gewöhnlichen Nomina gebildet werden.

(23.41) Ein Mensch studiert.

`(∃x0)(mensch(x0)∧studiert(x0))`

(23.42) Jeder Mensch studiert.

`(∀x0)(mensch(x0)→studiert(x0))`

Hierbei werden also die Quantoren (/`ein`/, /`jeder`/) aus der natürlichen Sprache durch eine doppelte Prädikation dargestellt: zum einen haben wir das gewöhnliche Nomen (hier /`Mensch`/), welches als 1-stellige Relation abgebildet wird, und dann das Verb, welches ebenfalls als 1-stellige Relation abgebildet wird. Man beachte die Junktoren ∧ und →. Man lese das erste etwa so: *es existiert ein x0, sodass x0 Mensch ist und x0 studiert*; das zweite lese man so: *jedes x0, sodass x0 ein Mensch ist, studiert*. Oder: *jedes x0, wenn x0 ein Mensch ist, dann studiert x0*. Natürlich lassen sich die Variablen wieder eliminieren. Zum Beispiel ist der letzte Satz gleichbedeutend mit *alles, welches ein Mensch ist, studiert*. Jedoch ist das Umsetzen eine Kunst, da die Elimination der Variablen genau die Mehrdeutigkeiten erzeugt, die man so gerne beseitigt hätte.

Auf diese Weise kann man sehen, dass beim Existenzquantor die Konjunktion und beim Allquantor die Implikation gesetzt werden muss. Die Wahl der Variablen ist unerheblich. Denn wenn ich behaupte, dass es ein A gibt, das B ist, so behaupte ich doch, dass es etwas gibt, das sowohl A als auch B ist. Und wenn ich behaupte, dass alle A B sind, so behaupte ich doch, dass alles, was existiert, wenn es A ist, so ist es ein B.

Bei einem transitiven Verb haben wir schon doppelt so viele Basissätze.

(23.43) Ein Kater sieht eine Maus.

`(∃x0)(kater(x0)∧(∃x1)(maus(x1)∧sieht(x0,x1)))`

(23.44) Ein Kater sieht jede Maus.

`(∃x0)(kater(x0)∧(∀x1)(maus(x1)→sieht(x0,x1)))`

(23.45) Jeder Kater sieht eine Maus.

`(∀x0)(kater(x0)→(∃x1)(maus(x1)∧sieht(x0,x1)))`

(23.46) Jeder Kater sieht jede Maus.

`(∀x0)(kater(x0)→(∀x1)(maus(x1)→sieht(x0,x1)))`

Man beachte hier, dass bei dem zweiten Quantor die Variable `x0` nicht mehr verwendet werden darf. Jede andere Variable tut es natürlich ebenso wie `x1`.

Zweitens füge ich hinzu, dass die Umsetzung in Prädikatenlogik mit Vorsicht zu genießen ist. So gibt es im Deutschen oft das Problem, dass Subjekt und Objekt nicht wirklich eindeutig bestimmbar sind. Hier ein Beispiel. Der folgende Satz kann als SVO oder als OVS gelesen werden. Entsprechend sieht die Übersetzung anders aus.

(23.47) Eine Ratte sieht jede Maus.
`(∃x0)(ratte(x0)∧(∀x1)(maus(x1)→sieht(x0,x1)))`
`(∀x0)(maus(x0)→(∃x1)(ratte(x1)→sieht(x0,x1)))`

Wir sagen in diesem Fall, dass (23.47) ambig ist. Die Prädikatenlogik hilft uns, diese Ambiguität zu diagnostizieren.

Übungen

[83] Es seien `angestellter` und `kunde` 2-stellige Relationssymbole, `bank`, `firma` und `bergsteiger` 1-stellige Relationssymbole; ferner seien `karl` und `elfriede` Konstanten und `direktor` und `hausmeister` 1-stellige Funktionssymbole. Ihre Bedeutung sei die der gleichlautenden umgangssprachlichen Wörter. Sind die folgenden Ausdrücke wohlgeformt? Wenn nein, warum nicht?

- `(∀x0)(angestellter(x0)→(∃x1)(firma(x1)∧hausmeister(x1,x0)))`
- `(∀x0)(hausmeister(x0)→(∃x1)(kunde(x1,x0)→firma(karl)))`
- `(∀x0)(∃x1)(firma(karl)`

[84] Was bedeuten folgende Formeln?

- `bergsteiger(karl)`
- `(∀x1)(bank(x1)→(∃x0)(bergsteiger(x0)∧kunde(x0,x1)))`
- `(∀x1)(bank(x1)→bergsteiger(direktor(x1)))`

[85] Übersetzen Sie in Prädikatenlogik:

- ”Elfriede ist Bergsteigerin.”
- “Jede Bank ist eine Firma.”
- “Wenn Karl Bergsteiger ist, so ist er Kunde einer Bank.”

[86] Nominale Ausdrücke wie “Hausmeister”, “Angestellter” und “Bergsteiger” werden in der obigen Sprache verschieden interpretiert. Ist das gerechtfertigt? Versuchen Sie, dies anhand von sprachlichen Daten zu rechtfertigen oder widerlegen wie zB

- “Karl ist Bergsteiger/Hausmeister/Angestellter.”
- “Karl ist *Bergsteiger/?Hausmeister/Angestellter der Firma X.”

Kapitel 24

Prädikatenlogik: Semantik I

Nun gehe ich zur Semantik der Prädikatenlogik über. Diese muss festlegen, was die Objekte sind, über die die Sprache redet. Das bedeutet, dass für jedes syntaktisch wohlgeformte Gebilde ein semantisches Korrelat bereitsteht. Dies ist für die Aussagen ein Wahrheitswert. Für die Terme hingegen sind es die sogenannten *Individuen*.

Zunächst eine allgemeine Definition.

Definition 24.1 (Struktur) *Es sei $\langle F, \Omega \rangle$ eine funktionale Signatur und $\langle G, \Psi \rangle$ eine relationale Signatur. Eine (Ω, Ψ)-**Struktur** ist ein Paar $\langle D, I \rangle$, wo D eine Menge (der **Objektbereich**, englisch domain) ist und I eine Funktion, welche auf $F \cup G$ erklärt ist und einem n-stelligen Relationssymbol eine n-stellige Relation auf D und einem n-stelligen Funktionssymbol eine n-stellige Funktion auf D zuordnet.*

Ich weise darauf hin, dass es nullstellige Funktionen und Relationen geben kann. Ein 0-stelliges Funktionssymbol wird durch eine Funktion $f : D^0 \to D$ interpretiert. Nun ist $D^0 = \{\varnothing\}$. Es ist also $\varnothing$ das einzige Argument, dem f etwas zuordnet; und es ist $f(\varnothing) \in D$. In dieser Form werde ich mich jedoch im Folgenden nicht ausdrücken. Sondern ich tue so, als seien die 0-stelligen Funktionen bereits Namen für diejenigen Objekte, die sie $\varnothing$ als Wert geben. So sage ich zum Beispiel, $\mathtt{0}$ sei der *Name* der Zahl Null oder bezeichne die Zahl Null, obwohl streng genommen $\mathtt{0}$ eine Funktion bezeichnet, die erst für $\varnothing$ den Wert 0 liefert.

Nullstellige Relationen sind Teilmengen von $D^0 = \{\varnothing\}$. Dies ist anders ausgedrückt die Zahl 1. Teilmengen sind $\varnothing$, also 0, und $\{\varnothing\}$, also 1. Diese Zahlen haben wir auch zu Wahrheitswerten erklärt. 0-stellige Relationen stellen sich also als Wahrheitswerte heraus, sodass die Konstanten dafür nur zwei Werte haben können. Deswegen betrachtet man 0-stellige Relationssymbole nur selten, weil sie nämlich durch die altbekannten Zeichen „`w`” und „`f`” repräsentiert werden und zweitens jederzeit definierbar sind. (Ich geben hier der Vollständigkeit halber an, dass zum Beispiel in Anwesenheit von Gleichheit /`(∃x0)x0=x0`/ die Funktion von „`w`” übernehmen kann und /`(¬(∃x0)x0=x0)`/ die Funktion von „`f`”.)

1. Beispiel. Es sei die Signatur sei $F := \{\mathtt{0}, 1, +, *, -\}$ mit $\Omega(\mathtt{0}) := \Omega(1) := 0$, $\Omega(-) := 1$, sowie $\Omega(+) := \Omega(*) := 2$ gegeben. Ferner sei $G := \{<\}$, $\Psi(<) := 2$. Dies definiert eine Sprache, die wir in dem vorigen Kapitel besprochen haben. Diese können wir auf sehr unterschiedliche Weise interpretieren. Ich gebe vier Strukturen an.

Die ganzen Zahlen, ganz normal. Sei $D_1 := \mathbb{Z}$, und $I_1(\mathtt{0})$ die Zahl 0, $I_1(1)$ die Zahl 1. (Eigentlich sind sie 0-stellige Funktionen, aber wir haben sie schlicht als Objekte erklärt.) $I_1(-)$ ordne jeder Zahl x die Zahl $-x$ zu. $I_1(+)$ sei die übliche Addition und $I_1(*)$ die Multiplikation. Diese Struktur heiße $\mathfrak{Z}$. Formal schreibt man dies wie folgt auf.

$$\begin{aligned} I_1(\mathtt{0}) &:= 0 \\ I_1(1) &:= 1 \\ I_1(-) &:= \{\langle x, -x\rangle : x \in \mathbb{Z}\} \\ I_1(+) &:= \{\langle x, y, x+y\rangle : x, y \in \mathbb{Z}\} \\ I_1(*) &:= \{\langle x, y, xy\rangle : x, y \in \mathbb{Z}\} \\ I_1(<) &:= \{\langle x, y\rangle : x < y\} \end{aligned} \tag{24.1}$$

Die natürlichen Zahlen, leicht verändert. Sei $D_2 := \mathbb{N}$, und $I_2(\mathtt{0})$ die Zahl 0, $I_2(1)$ die Zahl 1. $I_2(-)$ ordne jeder Zahl die 0 zu. $I_2(+)$ sei die übliche Addition, $I_2(*)$ die Multiplikation und $I_2(<)$ die übliche Kleinerrelation. Man beachte, dass wir wirklich nicht mehr verlangen, als dass $I_2(-)$ eine 1-stellige Funktion sei, und $I_2(+)$ und $I_2(*)$ zweistellige Funktionen sind, insofern könnte ich noch viel verrücktere Dinge hinschreiben. Diese Struktur heiße $\mathfrak{N}$.

Die Reste modulo 5. Sei $D_3 := \{0, 1, 2, 3, 4\}$, $I_3(\mathtt{0}) := 0$, $I_3(1) := 1$.

(24.2)

	$I_3(-)$
0	0
1	4
2	3
3	2
4	1

$I_3(+)$	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

$I_3(*)$	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

Und schließlich sei

$$I_3(<) := \{\langle 0,1\rangle, \langle 0,2\rangle, \langle 0,3\rangle, \langle 0,4\rangle, \langle 1,2\rangle, \langle 1,3\rangle, \langle 1,4\rangle, \langle 2,3\rangle, \langle 2,4\rangle, \langle 3,4\rangle\} \tag{24.3}$$

Diese Struktur heiße $\mathfrak{F}$.

2. Beispiel. Die nächste Signatur sei $F := \varnothing$, $G := \{\texttt{L1}, \texttt{L2}, \texttt{L3}, \texttt{L4}, \texttt{ben}\}$. Diese Sprache hat keine Funktionen, nur Relationen. Die Interpretation (nicht die einzig mögliche, natürlich) ist: D die die Menge der Straßen- und U-Bahnhaltestellen in Bielefeld. $J(\texttt{L1})$ ist die Menge der Haltestellen der Linie 1, $J(\texttt{L2})$ die Menge der Haltestellen der Linie 2, $J(\texttt{L3})$ die Menge der Haltestellen der Linie 3 und $J(\texttt{L4})$ die Menge der Haltestellen der Linie 4. Ferner ist $J(\texttt{ben})$ die Menge der Paare $\langle h, h'\rangle$, wo h und h' benachbarte Stationen auf einer Linie sind (egal welcher). Also haben wir

$$\begin{array}{rcl} J(\texttt{L1}) & = & \{\text{Schildesche, Sudbrackstraße, Hauptbahnhof}, \cdots\} \\ J(\texttt{L2}) & = & \{\text{Milse, Beckhausstraße, Hauptbahnhof}, \cdots\} \\ J(\texttt{L3}) & = & \{\text{Stieghorst, Jahnplatz, Hauptbahnhof}, \cdots\} \\ J(\texttt{L4}) & = & \{\text{Lohmannshof, Wellensiek, Hauptbahnhof}, \cdots\} \\ J(\texttt{ben}) & = & \{\langle\text{Schildesche, Heidegärten}\rangle, \\ & & \langle\text{Heidegärten, Schildesche}\rangle, \\ & & \langle\text{Hauptbahnhof, Jahnplatz}\rangle, \\ & & \langle\text{Jahnplatz, Hauptbahnhof}\rangle, \\ & & \langle\text{Lohmannshof, Wellensiek}\rangle, \\ & & \langle\text{Wellensiek, Lohmannshof}\rangle, \cdots\} \end{array} \tag{24.4}$$

Diese Struktur heiße $\mathfrak{B}$.

3. Beispiel. Die Ritter der Tafelrunde Die Tafelrunde bei Winchester hat 24 Plätze, dort saßen 24 Ritter, r_1 bis r_{24}. (Artus war nicht unter ihnen.) Unsere Sprache besitzt die folgenden Funktionen: `Gawain`, `Tristan`, `Iwein`, `Lanzelot`, `Parzival`, `L` („linker Nachbar von"), `R` („rechter Nachbar von"), `G` („Gegenübersitzender von"). Es ist $D := \{r_i : 1 \leq i \leq 24\}$.

$$\begin{array}{ll} I(\texttt{L}) & := \{\langle r_i, r_{i+1}\rangle : 1 \leq i \leq 23\} \cup \{\langle r_{24}, r_1\rangle\}, \\ I(\texttt{R}) & := \{\langle r_i, r_{i-1}\rangle : 2 \leq i \leq 24\} \cup \{\langle r_1, r_{24}\rangle\}, \\ I(\texttt{G}) & := \{\langle r_i, r_{12+i}\rangle : 1 \leq i \leq 12\} \cup \{\langle r_{12+i}, r_i\rangle : 1 \leq i \leq 12\} \end{array} \tag{24.5}$$

`Gawain` ist der Name von Ritter r_1, `Tristan` der Name von Ritter r_2, `Iwein` der Name von Ritter r_5, `Lancelot` der Name von Ritter r_{12}, `Parzival` der Name von Ritter r_{24}. Dazu kommen die Prädikate `H` („Held ohnegleichen”), welches einstellig ist, und `T` („rühmt die Taten von”), welches 2-stellig ist mit folgender Interpretation.

$$
\begin{array}{rcl}
I(\mathtt{H}) & := & \{r_1, r_3, r_5, r_{12}, r_{17}\} \\
I(\mathtt{T}) & := & \{\langle r_1, r_2\rangle, \langle r_1, r_3\rangle, \langle r_1, r_{11}\rangle, \langle r_2, r_1\rangle, \langle r_2, r_{12}\rangle \\
& & \langle r_3, r_1\rangle, \langle r_3, r_7\rangle, \langle r_3, r_1\rangle, \langle r_4, r_1\rangle, \langle r_5, r_1\rangle, \langle r_6, r_1\rangle, \\
& & \langle r_7, r_1\rangle, \langle r_8, r_1\rangle, \langle r_9, r_1\rangle, \langle r_{10}, r_1\rangle, \langle r_{11}, r_1\rangle, \langle r_{12}, r_1\rangle, \\
& & \langle r_{12}, r_{14}\rangle, \langle r_{13}, r_1\rangle, \langle r_{14}, r_1\rangle, \langle r_{14}, r_{13}\rangle, \\
& & \langle r_{15}, r_1\rangle, \langle r_{16}, r_1\rangle, \langle r_{17}, r_1\rangle, \langle r_{18}, r_1\rangle, \langle r_{19}, r_1\rangle, \langle r_{20}, r_1\rangle, \\
& & \langle r_{21}, r_1\rangle, \langle r_{22}, r_1\rangle, \langle r_{23}, r_1\rangle, \langle r_{24}, r_1\rangle\}
\end{array}
\tag{24.6}
$$

Diese Struktur heiße $\mathfrak{R}$.

Ich will nun angeben, wie man eine Formel in einer Struktur interpretiert. Dazu zunächst ein Begriff. Es sei Var die Menge der Variablen.

Definition 24.2 (Belegung) *Eine **Belegung** in einer Struktur $\mathfrak{M} = \langle D, I\rangle$ ist eine Funktion $\beta : \mathrm{Var} \to D$.*

Ist eine Belegung β und eine Struktur gegeben, so definieren wir jetzt für einen Term, was sein Wert unter der Belegung β in dieser Struktur ist.

$$
\begin{array}{ll}
[x]^{\mathfrak{M},\beta} & := \beta(x) \\
[f(t_0, \cdots, t_{n-1})]^{\mathfrak{M},\beta} & := I(f)([t_0]^{\mathfrak{M},\beta}, \cdots, [t_{n-1}]^{\mathfrak{M},\beta})
\end{array}
\tag{24.7}
$$

In der ersten Klausel steht x für eine Variable (zB `x2`). In der zweiten Klausel ist f ein n-stelliges Funktionssymbol und die t_i sind Terme.

Nehmen wir unsere erste Sprache, Interpretation 1. Die Struktur heiße $\mathfrak{Z}$. Die Belegung β erfülle $\beta(\mathtt{x0}) = 3$, $\beta(\mathtt{x1}) = -2$, die Belegung γ erfülle $\gamma(\mathtt{x0}) = 5$,

$\gamma(\texttt{x1}) = 1$. Dann ist

$$
\begin{array}{lrcl}
 & [\texttt{x0*x1}]^{\mathfrak{Z},\beta} & = & [\texttt{x0}]^{\mathfrak{Z},\beta} \cdot [\texttt{x1}]^{\mathfrak{Z},\beta} \\
 & & = & 3 \cdot (-2) = -6 \\
 & [\texttt{x0*x1}]^{\mathfrak{Z},\gamma} & = & [\texttt{x0}]^{\mathfrak{Z},\gamma} \cdot [\texttt{x1}]^{\mathfrak{Z},\gamma} \\
 & & = & 5 \cdot 1 = 5 \\
 & [\texttt{x0+(x1*(-x0))}]^{\mathfrak{Z},\beta} & = & [\texttt{x0}]^{\mathfrak{Z},\beta} + [\texttt{x1*(-x0)}]^{\mathfrak{Z},\gamma} \\
 & & = & [\texttt{x0}]^{\mathfrak{Z},\beta} + ([\texttt{x1}]^{\mathfrak{Z},\beta} \cdot [\texttt{-x0}]^{\mathfrak{Z},\beta}) \\
(24.8) & & = & [\texttt{x0}]^{\mathfrak{Z},\beta} + ([\texttt{x1}]^{\mathfrak{Z},\beta} \cdot (-[\texttt{x0}]^{\mathfrak{Z},\beta})) \\
 & & = & 3 + ((-2) \cdot (-3)) = 9 \\
 & [\texttt{x0+(x1*(-x0))}]^{\mathfrak{Z},\gamma} & = & [\texttt{x0}]^{\mathfrak{Z},\gamma} + [\texttt{x1*(-x0)}]^{\mathfrak{Z},\gamma} \\
 & & = & [\texttt{x0}]^{\mathfrak{Z},\gamma} + ([\texttt{x1}]^{\mathfrak{Z},\gamma} \cdot [\texttt{-x0}]^{\mathfrak{Z},\gamma}) \\
 & & = & [\texttt{x0}]^{\mathfrak{Z},\gamma} + ([\texttt{x1}]^{\mathfrak{Z},\gamma} \cdot (-[\texttt{x0}]^{\mathfrak{Z},\gamma})) \\
 & & = & 5 + (1 \cdot (-5)) = 0
\end{array}
$$

Jetzt nehmen wir uns die Struktur $\mathfrak{N}$ vor. Es sei jetzt $\beta(\texttt{x0}) = 2, \beta(\texttt{x1}) = 3$ (-3 darf als Wert ja nicht mehr vergeben werden), $\gamma(\texttt{x0}) := 5$, $\gamma(\texttt{x1}) := 1$. Es sei ferner $n(x) := 0$ für alle x.

$$
\begin{array}{lrcl}
 & [\texttt{x0*x1}]^{\mathfrak{N},\beta} & = & [\texttt{x0}]^{\mathfrak{N},\beta} \cdot [\texttt{x1}]^{\mathfrak{N},\beta} \\
 & & = & 2 \cdot 3 = 6 \\
 & [\texttt{x0*x1}]^{\mathfrak{N},\gamma} & = & [\texttt{x0}]^{\mathfrak{N},\gamma} \cdot [\texttt{x1}]^{\mathfrak{N},\gamma} \\
 & & = & 5 \cdot 1 = 5 \\
 & [\texttt{x0+(x1*(-x0))}]^{\mathfrak{N},\beta} & = & [\texttt{x0}]^{\mathfrak{N},\beta} + [\texttt{x1*(-x0)}]^{\mathfrak{N},\beta} \\
 & & = & [\texttt{x0}]^{\mathfrak{N},\beta} + ([\texttt{x1}]^{\mathfrak{N},\beta} \cdot [\texttt{-x0}]^{\mathfrak{N},\beta}) \\
(24.9) & & = & [\texttt{x0}]^{\mathfrak{N},\beta} + ([\texttt{x1}]^{\mathfrak{N},\beta} \cdot n([\texttt{x0}]^{\mathfrak{N},\beta})) \\
 & & = & 2 + (2 \cdot 0) = 2 \\
 & [\texttt{x0+(x1*(-x0))}]^{\mathfrak{N},\gamma} & = & [\texttt{x0}]^{\mathfrak{N},\gamma} + [\texttt{x1*(-x0)}] \\
 & & = & [\texttt{x0}]^{\mathfrak{N},\gamma} + ([\texttt{x1}]^{\mathfrak{N},\gamma} \cdot [\texttt{-x0}]^{\mathfrak{N},\gamma}) \\
 & & = & [\texttt{x0}]^{\mathfrak{N},\gamma} + ([\texttt{x1}]^{\mathfrak{N},\gamma} \cdot n([\texttt{x0}]^{\mathfrak{N},\gamma})) \\
 & & = & 5 + (1 \cdot 0) = 5
\end{array}
$$

Wir sehen also, dass der Wert sowohl von der Struktur selbst wie auch von der Belegung abhängt.

Als drittes Beispiel nehmen wir $\mathfrak{F}$, die Reste modulo 5. Die Belegungen sind

β : `x0` $\mapsto 2$, `x1` $\mapsto 3$, und γ : `x0` $\mapsto 0$, `x1` $\mapsto 1$.

$$
\begin{array}{lll}
[\texttt{x0*x1}]^{\mathfrak{F},\beta} & = & [\texttt{x0}]^{\mathfrak{F},\beta} \cdot [\texttt{x1}]^{\mathfrak{F},\beta} \\
 & = & 2 \cdot 3 = 1 \\
[\texttt{x0*x1}]^{\mathfrak{F},\gamma} & = & [\texttt{x0}]^{\mathfrak{F},\gamma} \cdot [\texttt{x1}]^{\mathfrak{F},\gamma} \\
 & = & 0 \cdot 1 = 0 \\
[\texttt{x0+(x1*(-x0))}]^{\mathfrak{F},\beta} & = & [\texttt{x0}]^{\mathfrak{F},\beta} + [\texttt{x1*(-x0)}]^{\mathfrak{F},\gamma} \\
 & = & [\texttt{x0}]^{\mathfrak{F},\beta} + ([\texttt{x1}]^{\mathfrak{F},\beta} \cdot [\texttt{-x0}]^{\mathfrak{F},\beta}) \\
 & = & [\texttt{x0}]^{\mathfrak{F},\beta} + ([\texttt{x1}]^{\mathfrak{F},\beta} \cdot (-[\texttt{x0}]^{\mathfrak{F},\beta})) \\
 & = & 2 + (2 \cdot (-3)) = 1 \\
[\texttt{x0+(x1*(-x0))}]^{\mathfrak{F},\gamma} & = & [\texttt{x0}]^{\mathfrak{F},\gamma} + [\texttt{x1*(-x0)}] \\
 & = & [\texttt{x0}]^{\mathfrak{F},\gamma} + ([\texttt{x1}]^{\mathfrak{F},\gamma} \cdot [\texttt{-x0}]^{\mathfrak{F},\gamma}) \\
 & = & [\texttt{x0}]^{\mathfrak{F},\gamma} + ([\texttt{x1}]^{\mathfrak{F},\gamma} \cdot (-[\texttt{x0}]^{\mathfrak{F},\gamma})) \\
 & = & 0 + (1 \cdot 0) = 0
\end{array}
\tag{24.10}
$$

Betrachten wir nun die Tafelrunde des König Artus. Sei β : `x0` $\mapsto r_3$, `x1` $\mapsto r_{15}$, so sind zum Beispiel $[\texttt{G(x0)}]^{\mathfrak{R},\beta}$ und $[\texttt{x1}]^{\mathfrak{R},\beta}$ identisch (nämlich r_{15}).

Zum Schluss noch ein Beispiel für die Struktur $\mathfrak{B}$. Sei β : `x0` $\mapsto$ Jahnplatz, `x1` $\mapsto$ Hauptbahnhof, `x2` $\mapsto$ Lohmannshof. Da wir keine Funktionssymbole haben, können wir außer den Variablen keine neuen Terme haben. Wir haben also lediglich solche banalen Tatsachen wie die Folgende zu vermelden.

$$[\texttt{x1}]^{\mathfrak{B},\beta} = \text{Hauptbahnhof} \tag{24.11}$$

Ich formuliere hier einen wichtigen Sachverhalt, den ich schon benutzt habe.

Lemma 24.3 (Koinzidenzlemma) *Es seien β und γ Belegungen in eine Struktur $\mathfrak{M}$. Ferner sei t ein Term. Ist dann $\beta(x) = \gamma(x)$ für alle in t auftretenden Variablen, so ist $[t]^{\mathfrak{M},\beta} = [t]^{\mathfrak{M},\gamma}$.*

Diese Tatsache kann man natürlich beweisen. Dies geschieht durch Induktion. Ist der Term eine Variable, dann gilt $[t]^{\mathfrak{M},\beta} = \beta(t) = \gamma(t) = [t]^{\mathfrak{M},\beta}$ nach Voraussetzung. Nun sei $t = f(u_0, \cdots, u_{n-1})$, wo f ein n-stelliges Funktionssymbol ist. Und es seien β und γ Belegungen, für die gilt: tritt x in t auf, so ist $\beta(x) = \gamma(x)$. Sei nun $i < n$ beliebig. Dann gilt, da jede Variable, die in u_i auftritt, auch in t auftritt, dass β und γ in allen in u_i auftretenden Variablen übereinstimmen. Nach Induktionsannahme ist dann

$$[u_i]^{\mathfrak{M},\beta} = [u_i]^{\mathfrak{M},\gamma} \tag{24.12}$$

Nun folgt

$$
\begin{array}{lrl}
 & & [t]^{\mathfrak{M},\beta} \\
 & = & [f(u_0, \cdots, u_{n-1})]^{\mathfrak{M},\beta} \\
 & = & I(f)([u_0]^{\mathfrak{M},\beta}, \cdots, [u_{n-1}]^{\mathfrak{M},\beta}) \\
(24.13) & = & I(f)([u_0]^{\mathfrak{M},\gamma}, \cdots, [u_{n-1}]^{\mathfrak{M},\gamma}) \\
 & = & [f(u_0, \cdots, u_{n-1}]^{\mathfrak{M},\gamma} \\
 & = & [t]^{\mathfrak{M},\gamma}
\end{array}
$$

Und das war zu zeigen.

Ein Term heißt **konstant**, falls in ihm keine Variablen auftreten. Solche Terme gibt es. So sind alle 0-stelligen Funktionen schon konstante Terme. Es folgt nun, dass der Wert konstanter Terme nicht von der Belegung abhängt. In gewissen Fällen kann man jedes Element des Bereichs durch einen konstanten Term benennen. So kann man jede natürliche Zahl bekommen, indem man von 0 aus jeweils Nachfolger bildet; man benötigt also nicht mehr als eine einzige 0-stellige Funktion mit Wert 0 und eine 1-stellige Funktion, die Nachfolgerfunktion. Ein konkretes Beispiel ist hier die Struktur $\mathfrak{R}$.

$$
\begin{array}{lll}
 & [\texttt{Gawain}]^{\mathfrak{R},\beta} & = r_1 \\
 & [\texttt{L(Gawain)}]^{\mathfrak{R},\beta} & = r_2 \\
(24.14) & [\texttt{L(L(Gawain))}]^{\mathfrak{R},\beta} & = r_3 \\
 & \cdots & \\
 & [\texttt{L}^{23}(\texttt{Gawain})]^{\mathfrak{R},\beta} & = r_{24} \\
 & [\texttt{L}^{24}(\texttt{Gawain})]^{\mathfrak{R},\beta} & = r_1
\end{array}
$$

Man kann also jeden Ritter finden, indem man von Gawain aus immer linke Nachbarn nimmt. Natürlich tut es außer r_1 jeder andere Ritter.

Übungen

[87] Es sei $\mathfrak{F}$ die Struktur der Reste modulo 5 und $\beta : \texttt{x0} \mapsto 3, \texttt{x1} \mapsto 1, \texttt{x2} \mapsto 4$. Bestimmen Sie

1. $[\texttt{(x0+1)*x2}]^{\mathfrak{F},\beta}$.

2. $[\texttt{(x0+x2)*(x0+x1)}]^{\mathfrak{F},\beta}$.

Es sei $\gamma : \texttt{x0} \mapsto 3, \texttt{x1} \mapsto 2, \texttt{x2} \mapsto 4$. Bestimmen Sie die Werte der Terme auch unter dieser Belegung.

[88] Geben Sie konstante Terme an, die die Zahlen von 0 bis 5 in $\mathfrak{F}$ bezeichnen.

[89] Ich behaupte, es gibt eine Zahl n derart, dass für alle Belegungen γ gilt: $[\mathtt{G(x0)}]^{\mathfrak{R},\gamma} = [\mathtt{L}^n\mathtt{(x0)}]^{\mathfrak{R},\gamma}$. Wenn ja, welches ist diese Zahl, wenn nein, geben Sie ein Beispiel, das diese Behauptung widerlegt.

[90] Geben Sie zunächst einen Term an, der den natürlichsprachigen Ausdruck möglichst wortgetreu wiedergibt und bestimmen Sie dann seinen Wert. Können Sie außer dem wortgetreuen Term einen einfacheren finden, der denselben Ritter bezeichnet?

1. *der Gegenüber des linken Nachbarn von Parzival*
2. *der linke Nachbar des Gegenübers des rechten Nachbarn von Iwein*
3. *der linke Nachbar des linken Nachbarn des rechten Nachbarn von Lancelot*

Kapitel 25

Prädikatenlogik: Semantik II

Wir führen die Definition der Interpretation weiter. Nachdem wir Termen einen Wert gegeben haben, werden wir nun auch den Formeln einen Wert geben. Zunächst eine Definition. Es sei β' eine x-**Variante** von β, falls $\beta'(y) = \beta(y)$ für alle $y \neq x$. Dabei darf $\beta'(x) = \beta(x)$ sein, muss es aber nicht. Wir schreiben $\beta' \sim_x \beta$, falls β' eine x-Variante von β ist. (In diesem Fall ist natürlich β auch eine x-Variante von β'; und β ist eine x-Variante von sich selbst.)

Definition 25.1 (Wahrheit) *Es sei $\mathfrak{M}$ eine Struktur und β eine Belegung.*

$$
\begin{array}{ll}
\langle \mathfrak{M}, \beta \rangle \vDash R(t_1, \cdots, t_n) & :\Leftrightarrow \langle [t_1]^{\mathfrak{M},\beta}, \cdots, [t_n]^{\mathfrak{M},\beta} \rangle \in I(R) \\
\langle \mathfrak{M}, \beta \rangle \vDash (t_1 = t_2) & :\Leftrightarrow [t_1]^{\mathfrak{M},\beta} = [t_2]^{\mathfrak{M},\beta} \\
\langle \mathfrak{M}, \beta \rangle \vDash (\neg \varphi) & :\Leftrightarrow \langle \mathfrak{M}, \beta \rangle \nvDash \varphi \\
\langle \mathfrak{M}, \beta \rangle \vDash (\varphi \wedge \chi) & :\Leftrightarrow \langle \mathfrak{M}, \beta \rangle \vDash \varphi \textit{ und } \langle \mathfrak{M}, \beta \rangle \vDash \chi \\
\langle \mathfrak{M}, \beta \rangle \vDash (\varphi \vee \chi) & :\Leftrightarrow \langle \mathfrak{M}, \beta \rangle \vDash \varphi \textit{ oder } \langle \mathfrak{M}, \beta \rangle \vDash \chi \\
\langle \mathfrak{M}, \beta \rangle \vDash (\varphi \rightarrow \chi) & :\Leftrightarrow \textit{wenn } \langle \mathfrak{M}, \beta \rangle \vDash \varphi \textit{ dann } \langle \mathfrak{M}, \beta \rangle \vDash \chi \\
\langle \mathfrak{M}, \beta \rangle \vDash (\forall x) \varphi & :\Leftrightarrow \textit{für alle } \beta' \sim_x \beta\textit{: } \langle \mathfrak{M}, \beta' \rangle \vDash \varphi \\
\langle \mathfrak{M}, \beta \rangle \vDash (\exists x) \varphi & :\Leftrightarrow \textit{für ein } \beta' \sim_x \beta\textit{: } \langle \mathfrak{M}, \beta' \rangle \vDash \varphi
\end{array}
\tag{25.1}
$$

Wir können nun in Analogie zu den Termen auch $[\varphi]^{\mathfrak{M},\beta}$ schreiben. Wir legen fest, dass genau dann $[\varphi]^{\mathfrak{M},\beta} = 1$ sein soll, wenn $\langle \mathfrak{M}, \beta \rangle \vDash \varphi$ und ansonsten sei $[\varphi]^{\mathfrak{M},\beta} := 0$. Dann ist $[\varphi]^{\mathfrak{M},\beta}$ offenkundig ein Wahrheitswert. Dann wird zum Beispiel

$$[\varphi \wedge \chi]^{\mathfrak{M},\beta} = [\varphi]^{\mathfrak{M},\beta} \cap [\chi]^{\mathfrak{M},\beta} \tag{25.2}$$

Sehen wir uns dies einmal an. Wir verwenden jetzt die 3. Interpretation, die Reste modulo 5. Diese Struktur haben wir $\mathfrak{F}$ genannt. Es sei eine Belegung β mit $\beta(\mathtt{x0}) =$

$1, \beta(\mathtt{x1}) = 3$ gegeben. Es ist

(25.3) $\langle \mathfrak{F}, \beta \rangle \vDash \mathtt{x0<(x1+1)}$

Denn $[\mathtt{x0}]^{\mathfrak{F},\beta} = 1$ und $[\mathtt{x1+1}]^{\mathfrak{F},\beta} = 4$, und $\langle 1, 4 \rangle \in I_3(<)$. Es ist aber

(25.4) $\langle \mathfrak{F}, \beta \rangle \nvDash \mathtt{x0<(x1+x1)}$

Denn $[\mathtt{x0}]^{\mathfrak{F},\beta} = 1$ und $[\mathtt{x1+x1}]^{\mathfrak{F},\beta} = 3 + 3 = 1$, und $\langle 1, 1 \rangle \notin I_3(<)$.

(25.5) $\langle \mathfrak{F}, \beta \rangle \vDash \mathtt{(\exists x0)(x0*x0*x0=1+1)}$

Dazu seien β_0, β_1, usf. $\mathtt{x0}$-Varianten von β, wobei $\beta_0(\mathtt{x0}) = 0$, $\beta_1(\mathtt{x0}) = 1$, usf. Dann ist (25.5) wahr, wenn

$$
(25.6)\qquad
\begin{array}{ll}
 & \langle \mathfrak{F}, \beta_0 \rangle \vDash \mathtt{x0*x0*x0=1+1} \\
\text{oder} & \langle \mathfrak{F}, \beta_1 \rangle \vDash \mathtt{x0*x0*x0=1+1} \\
\text{oder} & \cdots \\
\text{oder} & \langle \mathfrak{F}, \beta_4 \rangle \vDash \mathtt{x0*x0*x0=1+1}
\end{array}
$$

Nun ist $[\mathtt{1+1}]^{\mathfrak{F},\beta_0} = 2$ (und dies ist unabhängig von der Belegung), sowie $[\mathtt{x0*x0*x0}]^{\mathfrak{F},\beta_0} = 0 \cdot 0 \cdot 0 = 0$, $[\mathtt{x0*x0*x0}]^{\mathfrak{F},\beta_1} = 1 \cdot 1 \cdot 1 = 1$, $[\mathtt{x0*x0*x0}]^{\mathfrak{F},\beta_2} = 2 \cdot 2 \cdot 2 = 3$, $[\mathtt{x0*x0*x0}]^{\mathfrak{F},\beta_3} = 3 \cdot 3 \cdot 3 = 2$, $[\mathtt{x0*x0*x0}]^{\mathfrak{F},\beta_4} = 4 \cdot 4 \cdot 4 = 4$. Damit ist $[\mathtt{x0*x0*x0}]^{\mathfrak{F},\beta_3} = [\mathtt{1+1}]^{\mathfrak{F},\beta_3}$ und somit ist (25.5) wahr.

Es ist wichtig zu verstehen, dass hier der Name der Variablen hinter dem Quantor entscheidet, wo man die Belegung verändern darf, und dies wirkt sich selbstverständlich auf die Interpretation aus. Es ist zum Beispiel

(25.7) $\langle \mathfrak{F}, \beta \rangle \nvDash \mathtt{(\exists x3)(x0*x0*x0=1+1)}$

Denn die Formel in (25.7) ist wahr unter der Belegung β, wenn es eine $\mathtt{x3}$-Varianten β' von β gibt mit

(25.8) $\langle \mathfrak{F}, \beta' \rangle \vDash \mathtt{x0*x0*x0=1+1}$

Aber nun ist $\beta'(\mathtt{x0}) = \beta(\mathtt{x0}) = 1$, also ist $[\mathtt{x0*x0*x0}]^{\mathfrak{M},\beta'} = 1 \neq 2 = [\mathtt{1+1}]^{\mathfrak{M},\beta'}$.

Zum Abschluss werde ich noch zeigen, dass

(25.9) $\langle \mathfrak{F}, \beta \rangle \vDash \mathtt{(\forall x1)(\exists x0)x0*x0*x0=x1}$

Nach Definition ist dies der Fall, wenn für alle $\mathtt{x1}$-Varianten β' von β gilt

(25.10) $\langle \mathfrak{F}, \beta' \rangle \vDash \mathtt{(\exists x0)x0*x0*x0=x1}$

Dies wiederum ist der Fall, wenn es eine $\mathtt{x0}$-Variante β'' von β' gibt mit

(25.11) $\langle \mathfrak{F}, \beta'' \rangle \vDash \mathtt{x0*x0*x0=x1}$

1. Sei $\beta'(\texttt{x1}) = 0$. Dann setzen wir $\beta''(\texttt{x0}) = 0$ und erhalten $[\texttt{x0*x0*x0}]^{\mathfrak{F},\beta''} = 0$ und $[\texttt{x1}]^{\mathfrak{F},\beta''} = 0$.
2. Sei $\beta'(\texttt{x1}) = 1$. Dann setzen wir $\beta''(\texttt{x0}) = 1$ und erhalten $[\texttt{x0*x0*x0}]^{\mathfrak{F},\beta''} = 1$ und $[\texttt{x1}]^{\mathfrak{F},\beta''} = 1$.
3. Sei $\beta'(\texttt{x1}) = 2$. Dann setzen wir $\beta''(\texttt{x0}) = 3$ und erhalten $[\texttt{x0*x0*x0}]^{\mathfrak{F},\beta''} = 2$ und $[\texttt{x1}]^{\mathfrak{F},\beta''} = 2$.
4. Sei $\beta'(\texttt{x1}) = 3$. Dann setzen wir $\beta''(\texttt{x0}) = 2$ und erhalten $[\texttt{x0*x0*x0}]^{\mathfrak{F},\beta''} = 3$ und $[\texttt{x1}]^{\mathfrak{F},\beta''} = 3$.
5. Sei $\beta'(\texttt{x1}) = 4$. Dann setzen wir $\beta''(\texttt{x0}) = 4$ und erhalten $[\texttt{x0*x0*x0}]^{\mathfrak{F},\beta''} = 4$ und $[\texttt{x1}]^{\mathfrak{F},\beta''} = 4$.

Damit ist 25.9 bestätigt. Wir können dem auch eine anschaulichere Seite abgewinnen, wenn wir bedenken, dass „*" ja die Multiplikation der Reste modulo 5 bezeichnet. Die Formel $\varphi(x, y)$ definiert durch

$$\varphi(x, y) := x{*}x{*}x{=}y$$

besagt nichts anderes, als dass x die dritte Wurzel von y ist (oder eben y die dritte Potenz von x). Mithin besagt (25.9), dass jede Zahl in $\mathfrak{F}$ eine dritte Wurzel hat! Auf diese Weise sehen wir, dass neue Konzepte, die gar nicht im ursprünglichen Vokabular vorhanden waren, in der Prädikatenlogik definiert werden können.

Betrachten wir nun die Struktur des Bielefelder Liniennetzes. Sie heiße $\mathfrak{B}$. Es gilt darin zum Beispiel folgendes.

(25.12) $\langle\mathfrak{B},\beta\rangle \vDash \texttt{(}\forall\texttt{x0)(}\forall\texttt{x1)(ben(x0,x1)}\rightarrow\texttt{ben(x1,x0))}$

Denn sei β' eine $\texttt{x0}$-Variante von β. Dann gilt

(25.13) $\langle\mathfrak{B},\beta'\rangle \vDash \texttt{(}\forall\texttt{x1)(ben(x0,x1)}\rightarrow\texttt{ben(x1,x0))}$

Denn sei β'' eine $\texttt{x1}$-Variante von β'. Dann gilt

(25.14) $\langle\mathfrak{B},\beta''\rangle \vDash \texttt{ben(x0,x1)}\rightarrow\texttt{ben(x1,x0)}$

Und zwar sei $b = [\texttt{x0}]^{\mathfrak{B},\beta''}$ und $c = [\texttt{x1}]^{\mathfrak{B},\beta''}$. Dann ist, wenn $\langle b, c\rangle \in J(\texttt{ben})$ ist, ebenso auch $\langle c, b\rangle \in J(\texttt{ben})$. Denn es war ja $J(\texttt{ben})$ die Menge der Paare $\langle x, y\rangle$ derart, dass x und y benachbart sind. Diese Relation ist symmetrisch, das besagt gerade die Formel in (25.12).

Wenn man genau hinschaut, dann bemerkt man, dass die Belegung β eigentlich in (25.12) keine Rolle spielt. Dies gilt, weil die Formel keine freien Variablen enthält.

Es gilt offenkundig wieder eine Art Koinzidenzlemma. Wir definieren die freien Vorkommen einer Variable in einer Formel wie folgt.

Definition 25.2 (Freie Variable) *Es bezeichnet* $\mathrm{fv}(t)$ *die Menge der in einem Term* t *auftretenden Variablen. Ferner ist* $\mathrm{fv}(\varphi)$ *für eine Formel* φ *wie folgt definiert.*

(25.15)
$$\begin{array}{lcl} \mathrm{fv}(R(t_1, \cdots, t_n)) & := & \mathrm{fv}(t_1) \cup \mathrm{fv}(t_2) \cdots \cup \mathrm{fv}(t_n) \\ \mathrm{fv}(\neg\varphi) & := & \mathrm{fv}(\varphi) \\ \mathrm{fv}(\varphi \wedge \chi) & := & \mathrm{fv}(\varphi) \cup \mathrm{fv}(\chi) \\ \mathrm{fv}(\varphi \vee \chi) & := & \mathrm{fv}(\varphi) \cup \mathrm{fv}(\chi) \\ \mathrm{fv}(\varphi \rightarrow \chi) & := & \mathrm{fv}(\varphi) \cup \mathrm{fv}(\chi) \\ \mathrm{fv}((\forall x)\varphi) & := & \mathrm{fv}(\varphi) - \{x\} \\ \mathrm{fv}((\exists x)\varphi) & := & \mathrm{fv}(\varphi) - \{x\} \end{array}$$

Ist $\mathrm{fv}(\varphi) = \varnothing$*, so heißt* φ *ein* ***Satz****. Gilt* $\langle \mathfrak{M}, \beta \rangle \vDash \varphi$ *unabhängig von der Belegung, so schreiben wir auch* $\mathfrak{M} \vDash \varphi$*.*

Lemma 25.3 (Koinzidenzlemma II) *Sei* φ *eine Formel. Es sei* $\mathfrak{M}$ *eine Struktur und* β *und* γ *Belegungen derart, dass* $\beta(x) = \gamma(x)$ *für jedes* $x \in \mathrm{fv}(\varphi)$*. Dann ist* $\langle \mathfrak{M}, \beta \rangle \vDash \varphi$ *gdw.* $\langle \mathfrak{M}, \gamma \rangle \vDash \varphi$*. Ist* φ *insbesondere ein Satz, so gilt für je zwei Belegungen* β *und* γ*:* $\langle \mathfrak{M}, \beta \rangle \vDash \varphi$ *gdw.* $\langle \mathfrak{M}, \gamma \rangle \vDash \varphi$*.*

Wir haben also

(25.16) $\mathfrak{B} \vDash$ `(∀x0)(∀x1)(ben(x0,x1)→ben(x1,x0))`

Denn die Formel ist ein Satz, und wenn nur eine Belegung diesen erfüllt, dann erfüllen schon alle Belegungen diesen Satz.

Das Modell des Bielefelder Netzplans mag dürftig erscheinen. Zahlreiche Eigenschaften sind hier gar nicht angegeben. Das ist aber bei den meisten gar nicht nötig. Sie lassen sich nämlich aus den gegebenen Prädikaten definieren. Ich führe das an ein paar Beispielen vor. Das erste ist der Begriff des Endbahnhofs. Warum haben wir kein 1-stelliges Prädikat `endbhf` eingeführt, das wahr ist für alle und nur die Endbahnhöfe? Antwort: es kann wie folgt definiert werden:

(25.17) ε_1(`x0`) := `(∀x1)(∀x2)((L1(x0)∧L1(x1)∧L1(x2)`
`∧ben(x0,x1)∧ben(x0,x2))→x1=x2)`

Dies ist die Formel für Linie 1, analog ε_2 für Linie 2, ε_3 für Linie 3 und ε_4 für Linie 4.

Denn sei jetzt $\beta($`x0`$) = b$ für einen Bahnhof. Dann gilt $\langle \mathfrak{B}, \beta \rangle \vDash \varepsilon($`x0`$)$ genau dann, wenn b nur einen Nachbarn hat. (Probieren Sie das aus!) So ist dann die Formel ε_4 erfüllt für Lohmannshof aber nicht für Hauptbahnhof.

Eine andere Eigenschaft ist: auf der gleichen Linie liegen.

(25.18) `gl(x0,x1)` := `((L1(x0)∧L1(x1))∨(L2(x0)∧L2(x1))`
`∨(L3(x0)∧L3(x1))∨(L4(x0)∧L4(x1)))`

Auch dies muss man sich in Ruhe überlegen. Damit kann man nun definieren „sind benachbart und liegen auf der gleichen Linie".

Was wäre denn nun passiert, wenn wir, sagen wir mal, ein 1-stelliges Prädikat `endbhf` aufgenommen hätten? Dann hätten wir zunächst einmal ein neues Symbol, dessen Interpretation frei wählbar ist. Da das so nicht beabsichtigt war, sondern `endbhf` eine spezielle Bedeutung hat, die offenkundig vorhersagbar ist, müssen wir irgendwie erzwingen, dass `endbhf` auf bestimmte Weise interpretiert wird. Das wird im nächsten Kapitel thematisiert werden.

Übungen

[91] Prüfen Sie, ob die folgenden Sätze in der Struktur der Reste modulo 5 wahr sind.

1. `(∃x0)(x0*x0+x0=0)`
2. `(∀x0)(¬x0=0→x0*x0*x0*x0=1)`
3. `(∀x1)(x1=0∨x1+1=0∨x1+1+1=0∨x1+(-1)+(-1)=0∨x1+(-1)=0)`

[92] Sind die folgenden Sätze in der Struktur $\mathfrak{R}$ wahr? Übersetzen Sie diese zunächst in die formale Sprache.

1. „Es gibt einen Helden ohnegleichen, der einem Held ohnegleichen gegenüber sitzt."
2. „Jeder Held ohnegleichen rühmt die Taten von Gawain."

[93] Ist der folgende Satz wahr?

`(∀x0)(G(L(x0))=R(G(x0)))`

[94] Welche Belegungen erfüllen die folgende Formel?

$$H(L(x0))\wedge(\exists x1)(T(x0,G(x1))\rightarrow T(x1,x0))$$

Kapitel 26

Definierbarkeit

Die Formeln der Prädikatenlogik sind Beschreibungen der Strukturen, in denen sie interpretiert werden. Oft haben wir die Klasse der Strukturen, über die wir sprechen wollen, bereits definiert; diese ist meist nicht die Klasse *aller* möglichen Strukturen. Sondern sie ist eine Teilklasse. Dort gelten spezielle Beziehungen. So ist in der Struktur der Tafelrunde festgelegt, dass es nur 24 Plätze gibt. Das folgt aber aus nichts. Die Sprache legt uns darin nicht fest. Wir können die Prädikatenlogik nun dazu benutzen, solche Sachverhalte zu beschreiben. In diesem Kapitel soll es genau darum gehen, wie man dies zu Wege bringen kann.

Es ist eine kleine Kunst, die Intuition, die man hat, in die richtige Form zu bringen. So können wir die Tatsache, dass es nur 24 Plätze gibt, augenscheinlich mit der folgenden Formel beschreiben:

$$\rho_{24} = \texttt{(}\forall\texttt{x0)L}^{24}\texttt{(x0)=x0} \tag{26.1}$$

Diese Formel besagt, dass wir mit der Funktion „linker Nachbar" in 24 Schritten wieder an den Platz zurückkehren, von dem wir gestartet sind.

Wenn man nun etwas genauer nachdenkt, wird man sehen, dass die Intuition nicht ganz richtig ist. Es gibt zwei Probleme: wenn es *zwei* Tische gibt, so kann man die Ritter der einen Runde nicht von der anderen Runde aus erreichen, indem man der Funktion „linker Nachbar" folgt. Wir müssten also verlangen, dass dies nicht auftritt. Dazu bauen wir eine Klausel ein, die soviel sagt wie „jeder Ritter ist von jedem aus mit der Funktion *linker Nachbar* erreichbar". Diese abstrakte Tatsache kann man aber in der Prädikatenlogik leider nicht so ohne weiteres aufschreiben. In dem vorliegenden Fall aber kann man sich behelfen. Wir sagen, jeder Ritter ist in höchstens 24 Schritten aus von jedem anderen Ritter erreichbar:

$$\lambda_{24} = (\forall x0)(\forall x1)(x1{=}x0 \vee x1{=}L(x0) \vee x1{=}L(L(x0)) \vee \cdots \vee x1{=}L^{23}(x0)) \tag{26.2}$$

Ich gebe zu bedenken, dass ρ_{24} *nicht* aus λ_{24} folgt. λ_{24} ist auch dann erfüllt, wenn der Tisch 17 Plätze hat; in diesem Fall ist aber ρ_{24} falsch. Wir müssen schon fordern, dass beide gelten. Aber selbst dies reicht nicht! Denn es könnten ja auch nur 6 Ritter sein. Und so fahren wir fort, indem wir stest mehr Postulate hinzunehmen. Hat dieser Prozess ein Ende? Mit anderen Worten: lässt sich eine Struktur oder eine Menge von Strukturen mit Hilfe von endlich vielen Sätzen eindeutig festlegen? Darum soll es im Folgenden gehen.

Diese Übung ist nicht einfach nur ein Glasperlenspiel. Es ist integrativer Bestandteil wissenschaftlichen Arbeitens. Es kommt ja sehr oft vor, dass man gezwungen ist zu sagen, was ein gewisser Begriff bedeutet oder welche Eigenschaften gewisse Gegenstände haben (müssen). Diese Definitionen müssen (wenigstens theoretisch) exakt sein. Was also verlangt wird, ist eine Art „Formel" in der natürlichen Sprache, die alle und nur die Gegenstände erfüllen, die unter den Begriff fallen. Dass dies nicht gelingen kann, ohne eine genaue Untersuchung der Abhängigkeit vieler Begriffe untereinander zu starten, ist eines der Ergebnisse dieser Betrachtungen.

Ich komme noch einmal auf das Bielefelder Liniennetz zurück. Wir hatten eine Sprache angegeben und diese dann interpretiert. Nun ist alles eine Interpretation, welches den Relationssymbolen nur Mengen von Tupeln richtiger Bauart zuweist — und das ohne Rücksicht auf den höheren Zweck. Dass es sich dabei zum Beispiel um ein Liniennetz handelt, ist nirgendwo festgelegt. Die Namen wie /L1/ und /ben/ sind völlig willkürlich und frei wählbar.

Wir können nun mit Hilfe der Prädikatenlogik mittels gewisser Formeln eine gewisse Struktur „erzwingen". Wir halten die Formel fest und fragen, welche Strukturen sie erfüllen. So können wir zum Beispiel verlangen, dass die Interpretation des Relationszeichens /ben/ eine symmetrische Relation ist. Wie ich schon in der vorigen Vorlesung bemerkt hatte, wird die Symmetrie der Struktur durch die folgende Formel beschrieben.

$$\tau := (\forall x0)(\forall x1)(ben(x0,x1) \to ben(x1,x0)) \tag{26.3}$$

Dies bedeutet, dass eine *beliebige* Struktur $\mathfrak{M} = \langle D, I\rangle$ für unsere Signatur symmetrisch ist bezüglich I(ben), falls gilt

$$\mathfrak{M} \vDash \tau \tag{26.4}$$

Anders ausgedrückt: für jedes $\mathfrak{M} = \langle D, I\rangle$ ist $I(\mathtt{ben})$ genau dann symmetrisch, wenn $\mathfrak{M} \vDash \tau$. Da wir nun erwarten, dass Liniennetze symmetrisch sind, werden wir verlangen, dass nur noch Strukturen zugelassen sein sollen, die τ erfüllen. (Ich merke nur an, dass gelegentlich Bahnhöfe nur in einer Richtung benutzt werden können; einen solchen Fall habe ich London gesehen. Das wäre dann eine Verletzung der Symmetrie, sofern /ben(x0,x1)/ bedeutet, dass man von x0 aus eine einzige Station fahrend x1 erreichen kann — und nicht umgekehrt.)

Außerdem ist kein Bahnhof mit sich selbst benachbart.

$$\rho := \mathtt{(\forall x0)(\neg ben(x0,x0))} \tag{26.5}$$

Auch dies ist nicht selbstevident. Stellen Sie sich eine Vergnügungsbahn vor, die im Kreis fährt und bei der es lediglich einen Bahnhof gibt. Das Postulat schließt dies aus!

Drittens wollen wir, dass Bahnhöfe Nachbarn haben sollen. Genauer: ist a ein Bahnhof einer Linie, so gibt es noch einen Nachbarn auf der Linie. Dies schließt aus, dass wir Linien haben, auf denen man nicht fahren kann, oder Bahnhöfe, die nicht an Schienen liegen.

$$\mu_1 := \mathtt{(\forall x0)(L1(x0)\to(\exists x1)(L1(x1)\wedge ben(x0,x1)))}$$

Analoges verlangen wir von den anderen Linien. So bekommen wir also insgesamt vier Formeln, nämlich noch μ_2, μ_3 und μ_4.

Eine Linie soll sich aber auch nicht verzweigen. Das bedeutet, dass ein Bahnhof nur zwei Nachbarn haben darf. Die folgende Formel sagt dies auch, nur etwas anders: sie sagt, dass, wenn b, c und d Nachbarn von a sind, so sind zwei von diesen identisch.

$$\begin{aligned}\eta_1 := {} & \mathtt{(\forall x0)(\forall x1)(\forall x2)(\forall x3)(L1(x0)\wedge L1(x1)\wedge L1(x2)} \\ & \mathtt{\wedge L1(x3)\wedge ben(x0,x1)\wedge ben(x0,x2)\wedge ben(x0,x3)} \\ & \mathtt{\to(x1{=}x2\vee x1{=}x3\vee x2{=}x3))}\end{aligned} \tag{26.6}$$

Dann verlangen wir, dass es für eine Linie genau zwei Endhaltestellen geben soll:

$$\begin{aligned}\zeta_1 := {} & \mathtt{(\exists x0)(\exists x1)(}\varepsilon_1\mathtt{(x0)\wedge}\varepsilon_1\mathtt{(x1)\wedge(\neg x0{=}x1)\wedge(\forall x2)(}\varepsilon_1\mathtt{(x2)} \\ & \qquad\qquad\qquad\qquad \mathtt{\to(x0{=}x2\vee x1{=}x2)))}\end{aligned} \tag{26.7}$$

Hierbei ist ε_1 kein Prädikat der Sprache, es ist vielmehr definiert durch (25.17), welches ich hier wiederhole:

$$\begin{aligned}\varepsilon_1(x) := {} & \mathtt{(\forall x1)(\forall x2)(L1(}x\mathtt{)\wedge L1(x1)\wedge L1(x2)\wedge ben(}x\mathtt{,x1)} \\ & \qquad\qquad\qquad \mathtt{\wedge ben(}x\mathtt{,x2)\to x1{=}x2)}\end{aligned} \tag{26.8}$$

Man muss also in ζ_1 lediglich an die Stelle von $\varepsilon_1(x)$ die obige Formel einsetzen, wobei dann x durch die jeweils angezeigte Variable ersetzt werden muss (mehr davon im nächsten Kapitel).

Stellen wir uns nun die folgende Fragen: gegeben eine Struktur $\mathfrak{M} = \langle D, I\rangle$ derart, dass

(26.9) $$\mathfrak{M} \vDash \tau; \rho; \mu_1; \mu_2; \mu_3; \mu_4; \eta_1; \eta_2; \eta_3; \eta_4; \zeta_1; \zeta_2; \zeta_3; \zeta_4$$

Dann ist $I(\mathtt{L1})$ eine Teilmenge ℓ_1 von D, $I(\mathtt{ben})$ eine Teilmenge von $D \times D$, also eine 2-stellige Relation N auf D. Diese Relation muss dann symmetrisch sein, dh $a\,N\,b$ impliziert $b\,N\,a$ und irreflexiv: Es gilt nicht $a\,N\,a$. Ferner gibt es für ein $a \in D$ höchstens zwei $b, c \in D$ mit $a\,N\,b$ und $a\,N\,c$. Jetzt nehmen wir die Einschränkung N_1 von N auf ℓ_1. Wie sieht jetzt die Struktur von N_1 aus?

Wählen wir einfach mal einen Bahnhof aus, sagen wir a_0. Zwei Fälle treten auf (a) a_0 ist Endbahnhof, (b) a_0 ist kein Endbahnhof. Folgen wir erst einmal dem Fall (a). a_0 hat dann nur einen Nachbarn, nennen wir ihn a_1. a_1 ist entweder Endbahnhof, und hat nur einen Nachbarn (das muss dann a_0 sein), oder es hat zwei Nachbarn, von denen einer a_0 ist, der andere ist dann a_2. a_2 ist dann Endbahnhof mit a_1 als einzigen Nachbarn, oder hat zwei Nachbarn, einer davon ist a_1 und der zweite ist a_3. Und so weiter. So bekommen wir eine Kette a_0, a_1, a_2, und so weiter. Da wir endlich viele Bahnhöfe haben, kommen wir entweder zum Ende oder wir wiederholen uns. Im ersten Fall bekommen wir eine Kette

$$a_0, a_1, \cdots, a_n$$

in der a_0 und a_n Endbahnhöfe sind (und deswegen verschieden). Falls nun a_n einer der vorherigen Bahnhöfe ist, also $a_n = a_i$, so haben wir einen Kreis. Nun ist i sicher nicht 0, sonst wäre ja a_0 nicht Endbahnhof. Aber dann sind a_{i-1}, a_{i+1} sowie a_{n-1} jeweils Nachbarn von a_i und alle verschieden. Das war ausgeschlossen. Wir haben also mindestens mal eine Linie entdeckt, von a_0 nach a_n.

Schauen wir uns noch kurz den Fall (b) an. In diesem Fall könnte a_0 einfach mitten auf eine Linie liegen; oder aber, und dieser Fall ist nicht ausgeschlossen, wir haben einen Kreis: a_0, a_1, a_2, und so weiter bis a_{n-1}, wobei jetzt aber a_{n-1} Nachbar ist von a_0. In diesem Fall haben wir eine Ringlinie.

Dies ist nun leider nicht alles; die Menge ℓ_1 kann nun in eine Linie und eine ganze Reihe von Kreisen zerfallen (die allerdings keine Bahnhöfe gemeinsam haben). Das würde man gerne verhindern, man würde gerne eine Formel hinschreiben, die das verhindert. Die Prädikatenlogik ist aber leider nicht stark genug, um dies zu ermöglichen.

Was ich hier exemplarisch vorgemacht habe, heißt *Axiomatisierung* von Modellklassen.

Definition 26.1 *Eine Klasse $\mathcal{K}$ von Strukturen zu einer Signatur ist **axiomatisierbar**, falls es eine Menge Δ von Formeln gibt derart, dass $\mathfrak{M} \in \mathcal{K}$ genau dann, wenn $\mathfrak{M}$ alle Formeln von Δ erfüllt.*

Ist die Menge Δ endlich, so genügt bereits eine einzige Formel (man nehme die Konjunktion sämtlicher Formen in Δ).

Wir haben gesehen, dass man wesentliche Aspekte eines Liniennetzes axiomatisieren kann (aber nicht alle). Im Übrigen ist es möglich, eine Formel anzugeben, die nur dann erfüllt werden kann, wenn die Struktur im Wesentlichen die des Bielefelder Netzes ist (was das heißen soll, sage ich weiter unten). Dazu überlegt man sich, dass das Bielefelder Liniennetz endlich ist. Endliche Strukturen kann man in einer einzigen Formel beschreiben. Dazu zählen wir die Elemente auf, von e_0 bis e_{n-1}. (Sei etwa e_0 Lohmannshof, e_1 Wellensiek, usf.) Wir wählen Variable x_0 bis x_{n-1} (alle verschieden). Anschließend schreiben wir auf:

$$
(26.10) \qquad
\begin{array}{lcl}
\delta & := & \bigwedge_{i\neq j} \neg x_i{=}x_j \\
\pi_1 & := & \bigwedge_{e_i \text{ liegt auf Linie 1}} \mathtt{L1(}x_i\mathtt{)} \wedge \bigwedge_{e_i \text{ liegt nicht auf Linie 1}} \neg\mathtt{L1(}x_i\mathtt{)} \\
\pi_2 & := & \bigwedge_{e_i \text{ liegt auf Linie 2}} \mathtt{L2(}x_i\mathtt{)} \wedge \bigwedge_{e_i \text{ liegt nicht auf Linie 2}} \neg\mathtt{L2(}x_i\mathtt{)} \\
\pi_3 & := & \bigwedge_{e_i \text{ liegt auf Linie 3}} \mathtt{L3(}x_i\mathtt{)} \wedge \bigwedge_{e_i \text{ liegt nicht auf Linie 3}} \neg\mathtt{L3(}x_i\mathtt{)} \\
\pi_4 & := & \bigwedge_{e_i \text{ liegt auf Linie 4}} \mathtt{L4(}x_i\mathtt{)} \wedge \bigwedge_{e_i \text{ liegt nicht auf Linie 4}} \neg\mathtt{L4(}x_i\mathtt{)} \\
\nu & := & \quad \bigwedge_{e_i \text{ und } e_j \text{ sind benachbart}} \mathtt{ben(}x_i\mathtt{,}x_j\mathtt{)} \\
 & & \wedge \bigwedge_{e_i \text{ und } e_j \text{ sind nicht benachbart}} \neg\mathtt{ben(}x_i\mathtt{,}x_j\mathtt{)}
\end{array}
$$

Der gesuchte Satz ist dann:

$$
(26.11) \qquad
\begin{array}{l}
\Delta_{\mathfrak{B}} := \mathtt{(}\exists x_0\mathtt{)(}\exists x_1\mathtt{)}\cdots\mathtt{(}\exists x_{n-1}\mathtt{)}\pi_1\wedge\pi_2\wedge\pi_3\wedge\pi_4\wedge\nu \\
\qquad\qquad\qquad \wedge\mathtt{(}\forall x_n\mathtt{)(}x_0{=}x_n\vee x_1{=}x_n\vee\cdots\vee x_{n-1}{=}x_n\mathtt{))}
\end{array}
$$

Dieser Satz sagt: Es gibt n Bahnhöfe (etwa e_0 bis e_{n-1}, aber nicht notwendig diese), sie sind verschieden und liegen wie angegeben auf den Linien 1 bis 4, und sind wie angegeben benachbart, und es gibt keinen weiteren Bahnhof. (Genauer gesagt besagt die letzte Teilformel dieses: Welcher Wert auch immer für x_n gegeben wird, er muss einer von e_0 bis e_{n-1} sein. Damit haben wir genau n Bahnhöfe.) Die Formel $\Delta_{\mathfrak{B}}$ heißt auch das *Diagramm* der Struktur $\mathfrak{B}$. Sie listet alle positiven wie negativen Elementarfakten auf.

Dies bedeutet aber nicht, dass nur das Bielefelder Liniennetz diese Formel erfüllt. Die Formel kann nämlich nichts über die tatsächliche Identität der beteiligten

Objekte sagen, nur über ihre Beziehung untereinander. Sind $\mathfrak{M}$ und $\mathfrak{B}$ strukturähnlich (wir sagen isomorph), dann können sie nicht durch Formeln unterschieden werden.

Definition 26.2 *Es seien $\mathfrak{M} = \langle D, I\rangle$ und $\mathfrak{M}' = \langle D', I'\rangle$ zwei Strukturen. $\mathfrak{M}$ heißt* ***isomorph*** *zu $\mathfrak{M}'$, in Zeichen $\mathfrak{M} \cong \mathfrak{M}'$, falls es eine bijektive Funktion $h : D \to D'$ gibt mit folgender Eigenschaft: Für jedes n und jede n-stellige Relation r ist*

(26.12) $\langle a_0, a_1, \cdots, a_{n-1}\rangle \in I(r) \Leftrightarrow \langle h(a_0), h(a_1), \cdots, h(a_{n-1})\rangle \in I'(r)$

und für jede n-stellige Funktion f:

(26.13) $h(I(f)(a_0, a_1, \cdots, a_{n-1})) = I'(f)(h(a_0), h(a_1), \cdots, h(a_{n-1}))$

Erinnern wir uns an das Beispiel der Reste modulo 5. Es war $D_3 = \{0, 1, 2, 3, 4\}$, $I_3(\mathbb{0}) = 0$, $I_3(1) = 1$.

(26.14)

	$I_3(-)$
0	0
1	4
2	3
3	2
4	1

$I_3(+)$	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

$I_3(*)$	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

Und außerdem war

(26.15) $I_3(<) := \{\langle 0, 1\rangle, \langle 0, 2\rangle, \langle 0, 3\rangle, \langle 0, 4\rangle, \langle 1, 2\rangle, \langle 1, 3\rangle, \langle 1, 4\rangle, \langle 2, 3\rangle,$
$\langle 2, 4\rangle, \langle 3, 4\rangle\}$

Ich gebe nun eine andere Struktur. Sei $E := \{\blacktriangle, \mho, \varnothing, \sqsupset, \bowtie\}$, $J(\mathbb{0}) := \blacktriangle$, $J(1) := \mho$.

(26.16)

	$J(-)$
$\blacktriangle$	$\blacktriangle$
$\mho$	$\bowtie$
$\varnothing$	$\sqsupset$
$\sqsupset$	$\varnothing$
$\bowtie$	$\mho$

$J(+)$	$\blacktriangle$	$\mho$	$\varnothing$	$\sqsupset$	$\bowtie$
$\blacktriangle$	$\blacktriangle$	$\mho$	$\varnothing$	$\sqsupset$	$\bowtie$
$\mho$	$\mho$	$\varnothing$	$\sqsupset$	$\bowtie$	$\blacktriangle$
$\varnothing$	$\varnothing$	$\sqsupset$	$\bowtie$	$\blacktriangle$	$\mho$
$\sqsupset$	$\sqsupset$	$\bowtie$	$\blacktriangle$	$\mho$	$\varnothing$
$\bowtie$	$\bowtie$	$\blacktriangle$	$\mho$	$\varnothing$	$\sqsupset$

$J(*)$	$\blacktriangle$	$\mho$	$\varnothing$	$\sqsupset$	$\bowtie$
$\blacktriangle$	$\blacktriangle$	$\blacktriangle$	$\blacktriangle$	$\blacktriangle$	$\blacktriangle$
$\mho$	$\blacktriangle$	$\mho$	$\varnothing$	$\sqsupset$	$\bowtie$
$\varnothing$	$\blacktriangle$	$\varnothing$	$\bowtie$	$\mho$	$\sqsupset$
$\sqsupset$	$\blacktriangle$	$\sqsupset$	$\mho$	$\bowtie$	$\varnothing$
$\bowtie$	$\blacktriangle$	$\bowtie$	$\sqsupset$	$\varnothing$	$\mho$

Und schließlich sei

$$J(<) := \{\langle \blacktriangle, \mho\rangle, \langle \blacktriangle, \varnothing\rangle, \langle \blacktriangle, \sqsupset\rangle, \langle \blacktriangle, \bowtie\rangle, \langle \mho, \varnothing\rangle, \langle \mho, \sqsupset\rangle, \langle \mho, \bowtie\rangle, \langle \varnothing, \sqsupset\rangle, \langle \varnothing, \bowtie\rangle, \langle \sqsupset, \bowtie\rangle\} \tag{26.17}$$

Es ist $\langle E, J\rangle \cong \langle D, I_3\rangle$. Und zwar ist h definiert durch $h : 0 \mapsto \blacktriangle$, $1 \mapsto \mho$, $2 \mapsto \varnothing$, $3 \mapsto \sqsupset$, $4 \mapsto \bowtie$ ein Isomorphismus.

Dazu muss man viele Einzelheiten prüfen. Gehen wir ein paar Fälle durch. Es ist $I_3(-)(4) = 1$. Nun ist $h(4) = \bowtie$ und $h(1) = \mho$. Also muss $J(-(\bowtie)) = \mho$ sein. Das ist in der Tat der Fall. Ebenso muss

$$h(I_3(*)(2, 3)) = J(*)(h(2), h(3)) \tag{26.18}$$

sein. Es ist $h(2) = \varnothing$, $h(3) = \sqsupset$. $I_3(*)(2, 3) = 1$ und $h(1) = \mho$. Also muss $J(\varnothing, \sqsupset) = \mho$ sein, und das ist der Fall.

Satz 26.3 *Es sei $h : D \to D'$ ein Isomorphismus von $\mathfrak{M} = \langle D, I\rangle$ auf $\mathfrak{M}' = \langle D', I'\rangle$.*

① *Ist $\beta : \mathrm{Var} \to D$ eine Belegung, so ist $h \circ \beta : \mathrm{Var} \to D'$ eine Belegung, und es ist*

$$\langle\langle D, I\rangle, \beta\rangle \vDash \varphi \Leftrightarrow \langle\langle D', I'\rangle, \beta'\rangle \vDash \varphi$$

② *Ist $\beta' : \mathrm{Var} \to D'$ eine Belegung, so ist $h^{-1} \circ \beta' : \mathrm{Var} \to D$ eine Belegung, und es ist*

$$\langle\langle D, I\rangle, \beta\rangle \vDash \varphi \Leftrightarrow \langle\langle D', I'\rangle, \beta'\rangle \vDash \varphi$$

③ *Genau dann ist $\mathfrak{M} \vDash \varphi$, wenn $\mathfrak{M}' \vDash \varphi$.*

Dieser Satz ist nicht schwer zu zeigen. Es sagt uns, dass wir keine Hoffnung haben, zwischen isomorphen Strukturen unterscheiden zu können.

Es sei nicht unerwähnt, dass wir auf dem Wege der Beschreibung von Strukturen uns auch noch eines anderen Mittels bedient haben, das ich kurz vorstellen will. Auch wenn in der formalen Sprache gewisse Konzepte nicht vorkommen, so kann man sie dennoch *definieren*. An einigen Stellen ist dieser Begriff schon gefallen. Ich will ihn noch formal einführen.

Definition 26.4 (Definierbarkeit) *Es sei $\mathfrak{M}$ eine Struktur und $A \subseteq M$ eine beliebige Teilmenge. A heißt **definierbar** in $\mathfrak{M}$, falls es eine Formel $\varphi(x)$ mit einer einzigen freien Variable x gibt, sodass $\langle \mathfrak{M}, \beta\rangle \vDash \varphi(x)$ genau dann gilt, wenn $\beta(x) \in A$.*

So ist etwa der Begriff „Primzahl” in $\mathbb{N}$ definierbar, wenn man als Grundoperation nur die Multiplikation und die Konstante 1 zur Verfügung hat. Denn wir können sagen, dass p genau dann eine Primzahl ist, wenn $p \neq 1$ und wenn für je zwei Zahlen m, n gilt: Ist $p = mn$, dann ist $p = m$ oder $p = n$. Dies lässt sich in der Prädikatenlogik aufschreiben:

(26.19) `(1<x0)∧(∀x1)(∀x2)(x0=x1*x2→(x0=x1∨x0=x2))`

Diese Formel definiert also die Eigenschaft, dass `x0` Primzahl ist. Ebenso haben wir den Begriff „Endbahnhof” im Liniennetz Bielefelds definieren können.

Die Definition der Definierbarkeit lässt sich auf beliebige n-stellige Relationen ausdehnen (man ersetze dann φ durch $\chi(x_0, \cdots, x_{n-1})$). Hier ist ein Beispiel. Die Relation “kleiner oder gleich” lässt sich in $\mathfrak{N}$ wie folgt definieren:

(26.20) `(∃x2)(x0+x2=x1)`

Dies funktioniert aber nicht in $\mathfrak{Z}$. Dennoch ist die Relation definierbar. Dazu muss man allerdings tief in die Trickkiste greifen. In der Zahlentheorie wird der folgende Satz gezeigt: *Jede natürliche Zahl ist Summe von vier Quadratzahlen*. Da Quadratzahlen immer positiv sind, können wir anstelle der vorigen Definition die folgende hinschreiben:

(26.21) `(∃x2)(∃x3)(∃x4)(∃x5)(x0+x2*x2+x3*x3+x4*x4+x5*x5=x1)`

Diese Definition funktioniert auch in $\mathfrak{N}$, ist aber komplizierter als nötig.

Ebenso kann man den Begriff der Definierbarkeit auf Funktionen ausdehnen.

Definition 26.5 (Definierbarkeit) *Es sei $\mathfrak{M}$ eine Struktur und $f : D \to D$ eine beliebige Funktion. f heißt **definierbar**, falls es einen Term $t(x)$ mit einer einzigen Variable x gibt, sodass $\langle \mathfrak{M}, \beta \rangle \vDash y = t(x)$ genau dann gilt, wenn $\beta(y) = f(\beta(x))$.*

So ist die Funktion „übernächster linker Nachbar von” in $\mathfrak{R}$ definierbar durch `L(L(x0))`.

Übungen

[95] Es sei $\mathfrak{M} = \langle D, I \rangle$ eine endliche Struktur. Es gebe für jedes Element m ein konstanter Term t_m, dessen Interpretation $[t_m]^{\mathfrak{M}} = m$. Zeigen Sie, dass jede Teilmenge von D definierbar ist. (Anmerkung: es ist sogar jede n-stellige Relation definierbar.)

[96] Es sei eine Sprache mit nur einem einstelligen Funktionszeichen s gegeben. Wir wollen die Theorie der Struktur $\mathfrak{D}$, bestehend aus dem Bereich $D = \{a, b, c\}$ und der Funktion $I(s) = \{\langle a, b\rangle, \langle b, c\rangle, \langle c, a\rangle\}$ axiomatisieren. Dazu reichen die folgenden Aussagen:

1. Es gibt genau drei Elemente.
2. Die Funktion $I(s)$ ist bijektiv.
3. $I(s)$ beschreibt einen Kreis der Länge 3.

Schreiben Sie je einen Satz in der Prädikatenlogik auf, der dies ausdrückt.

[97] Das Diagramm der Struktur ist wie folgt:

$$\mathtt{(\exists x0)(\exists x1)(\exists x2)}\delta \tag{26.22}$$

wobei δ wie folgt aussieht:

$$\begin{array}{ll} & \mathtt{\neg x0{=}x1 \wedge \neg x0{=}x2 \wedge \neg x1{=}x2} \\ \wedge & \mathtt{(\forall x3)(x3{=}x0 \vee x3{=}x1 \vee x3{=}x2)} \\ \wedge & \mathtt{s(x0){=}x1 \wedge s(x1){=}x2 \wedge s(x2){=}x0} \end{array} \tag{26.23}$$

Benennen Sie *alle* Belegungen in $\mathfrak{D}$, die δ erfüllen (= wahr machen).

[98] Zeigen Sie, dass die Menge der Potenzen von 2 in $\mathfrak{N}$ definierbar ist. *Anleitung.* Man kann die Eigenschaft, Potenz von 2 zu sein, nicht direkt in Prädikatenlogik aufschreiben. Es ist aber k genau dann eine Potenz von 2, wenn sie nur 2 als Primteiler besitzt.

Kapitel 27

Substitution

In diesem Kapitel will ich noch einmal auf die Frage nach der Ersetzung zurückkommen. In 8 hatte ich schon den Begriff der Zeichenkettenersetzung vorgestellt. Dies ist die Ersetzung eines Teilworts durch eine andere Zeichenkette. Ist etwa eine Zeichenkette $\vec{x}$ gegeben und darin ein Vorkommen der Zeichenkette $\vec{y}$, so ersetzen wir dieses Vorkommen durch $\vec{z}$, indem wir $\vec{y}$ an dieser Stelle herausnehmen und $\vec{z}$ dort einsetzen. Seien zum Beispiel folgende Zeichenketten gegeben:

$$\vec{x} := \texttt{((w→(¬w))∨w)}, \qquad \vec{y} := \texttt{(¬w)}, \qquad \vec{z} := \texttt{(¬(¬w))} \tag{27.1}$$

Dann ist das Vorkommen von $\vec{y}$ eindeutig bestimmt, und die Ersetzung geht wie folgt vor sich:

$$\begin{array}{ccc} \texttt{((w→} & \underline{\texttt{(¬w)}} & \texttt{)∨w)} \\ & \downarrow & \\ \texttt{((w→} & \texttt{(¬(¬w))} & \texttt{)∨w)} \end{array} \tag{27.2}$$

In dem gegebenen Fall ist wie gesagt das Vorkommen von $\vec{y}$ eindeutig. Das muss nicht so sein. Sei zum Beispiel $\vec{y} := \texttt{w}$. Dann gibt es insgesamt drei Möglichkeiten der Ersetzung, je nachdem, welches Vorkommen wir wählen.

$$\begin{array}{ccc} \texttt{((} & \underline{\texttt{w}} & \texttt{→(¬w))∨w)} \\ & \downarrow & \\ \texttt{((} & \texttt{(¬(¬w))} & \texttt{→(¬(¬w)))∨w)} \end{array} \tag{27.3}$$

$$\begin{array}{ccc} \texttt{((w→(¬} & \underline{\texttt{w}} & \texttt{))∨w)} \\ & \downarrow & \\ \texttt{((w→(¬} & \texttt{(¬(¬w))} & \texttt{)∨w)} \end{array} \tag{27.4}$$

(27.5)
$$\begin{array}{lcl} \mathtt{((w{\to}(\neg w))\vee} & \underline{\mathtt{w}} & \mathtt{)} \\ & \downarrow & \\ \mathtt{((w{\to}(\neg w))\vee} & \mathtt{(\neg(\neg w))} & \mathtt{)} \end{array}$$

In der Logik werden wir oft die Notation $[\vec{z}/\vec{y}]\vec{x}$ für das Ergebnis der Ersetzung von $\vec{y}$ durch $\vec{z}$ sehen. (Ich habe diese Notation auch schon benutzt.) Dies ist jedoch nur dann zulässig, wenn klargestellt wird, wie das gemeint ist. Eine Variante ist, dass man sagt, es soll ein *beliebiges Vorkommen* von $\vec{y}$ fixiert und durch $\vec{z}$ ersetzt werden. Die andere ist, dass man *alle Vorkommen zugleich ersetzt*. Dies sähe in dem vorliegenden Fall so aus.

(27.6)
$$\begin{array}{lclclcl} \mathtt{((} & \underline{\mathtt{w}} & \mathtt{{\to}(\neg} & \underline{\mathtt{w}} & \mathtt{))\vee} & \underline{\mathtt{w}} & \mathtt{)} \\ & \downarrow & & \downarrow & & \downarrow & \\ \mathtt{((} & \mathtt{(\neg(\neg w))} & \mathtt{{\to}(\neg} & \mathtt{(\neg(\neg w))} & \mathtt{))\vee} & \mathtt{(\neg(\neg w))} & \mathtt{)} \end{array}$$

In der Praxis darf deswegen ein Zusatz, welche Ersetzung gemeint ist, nicht fehlen. In vielen Softwaresystemen kann man in Texten suchen und ersetzen; dort gibt es zwei Optionen (wenn nicht sogar mehr): entweder das *nächste Vorkommen* wird ersetzt oder alle. Man muss das explizit festlegen.

Mit dieser Ersetzung gibt es einige Probleme, über die man unbedingt Bescheid wissen muss. Der Intention nach ist die Ersetzung ja nicht eine Ersetzung von Zeichenketten durch Zeichenketten. Sondern es ist eine Ersetzung von Zeichenketten, die im Aufbau der Zeichenkette tatsächlich gebildet werden. Dazu bedienen wir uns der Grammatik, die ja die Konstituenten erst definiert.

Das betrifft den Fall von `w∨f∨w`, wo wir vereinbart hatten, dass der Ausdruck linksgeklammert gelesen werden soll. Dies bedeutet, dass er aus `w∨f` und `w` zusammengesetzt ist und nicht aus `w` und `f∨w`. Deswegen kommt `f∨w` zwar als *Zeichenkette* vor, nicht aber als *Konstituente*.

Die Grammatik reicht allerdings nicht ganz hin, aus Gründen, die ich nicht weiter ausführen werde. Wirklich relevant sind nämlich die Klauseln des Zeichensystems. In der Aussagenlogik wird dies offensichtlich. Wir hatten die Bedeutung einer Aussage, etwa `/(p0∨p10)/`, als die Menge der Belegungen gedeutet, die sie erfüllen. Es ist also

(27.7) $$[\mathtt{(p0\vee p10)}] = \{\beta : \beta(\mathtt{p0}) = 1 \text{ oder } \beta(\mathtt{p10}) = 1\}$$

Wir bekommen dies, indem wir die Definition von $[-]$ anschauen. Es ist ja

(27.8) $$\beta(\mathtt{(p0\vee p10)}) = 1 \text{ gdw. } \beta(\mathtt{p0}) = 1 \text{ oder } \beta(\mathtt{p10}) = 1$$

Damit offenbart sich /`p0`/ wie auch /`p10`/ als Teil von /`(p0∨p10)`/. Hier allerdings hört der Prozess auf: Es gibt keine kleineren Teile mehr. Insbesondere ist /`p1`/ nicht Teil von /`p10`/! Und das, obwohl es ganz so aussieht, wie eine Aussage.

Dieser Punkt ist enorm wichtig. Es gibt ganz analoge Probleme in der natürlichen Sprache. Es ist ja /`Klaus`/ nicht Bestandteil von /`Klaustrophobie`/, ebenso wie /`ich`/ nicht Bestandteil von /`Eichhörnchen`/ ist. Wäre es nämlich nicht so, so könnte ich /`Klaus`/ durch /`Peter`/ ersetzen und bekäme /`Petertrophobie`/, welches kein Wort des Deutschen ist. Ebenso ergäbe die Ersetzung von /`ich`/ durch /`du`/ in /`Eichhörnchen`/ das Wort /`Eduhörnchen`/, ebenfalls kein sinnvolles Ergebnis.

Dies scheint nun sehr offensichtlich zu sein, aber man muss schon gut aufpassen, um die Begründung zu verstehen: Teil ist das, was im Prozess der Erzeugung des Ganzen auch tatsächlich vorkommen wird (und zwar an der Stelle, wo wir das Vorkommen markiert haben). Da ich für das Deutsche kein solches System angeben werde, hängen die Beispiele jetzt natürlich in der Luft. In den logischen Sprachen dagegen ist alles bereits festgelegt, und wir können die Definitionen genau nachvollziehen.

In der Aussagenlogik von Kapitel 17 ist also $\vec{y}$ eine Teilaussage von $\vec{x}$ wenn sie entweder von Klammern eingerahmt ist (also die Form $\texttt{(}\vec{u}\texttt{)}$ hat) oder die Form $\texttt{p}\vec{u}$ hat, wobei $\vec{u}$ so lang ist, wie es in $\vec{x}$ möglich ist. (Das heißt konkret, das nächste Zeichen nach dem Vorkommen von $\vec{u}$ darf keine Ziffer sein.) Man darf also nicht aus /`p123`/ die Folge /`p1`/ oder /`p12`/ auswählen; diese sind Teilzeichenketten, aber keine *Teilformeln*. Ersetzung ist nun stets Ersetzung von *Teilformelvorkommen* durch *Formeln*, oder in diesem Fall, von Variablen durch Aussagen, genauer gesagt, von Zeichenketten, die *als Variable vorkommen*. Die Folge /`p0`/ kommt in /`p01`/ nicht als Variable vor; die Grammatik lässt die Teilkonstituente /`p0`/ in /`p01`/ nicht zu, obwohl natürlich /`p0`/ selbst auch eine Ableitung hat. Diese Ableitung ist aber nicht Teil einer Ableitung von /`p01`/.

Ich komme jetzt auf die Prädikatenlogik zu sprechen. Nehmen wir die Formel (26.7). Diese Formel ist lediglich schematisch gegeben. Darin kommt dreimal ein Ausdruck $\varepsilon_1(x)$ vor mit einer schematischen Variablen x. Eine schematische Variable (oder auch **Metavariable**) ist ein Platzhalter für eine beliebige Variable, die man im Einzelfall festlegen darf (und muss). Ich gebe diese Formel hier an.

$$\varepsilon_1(x) = \texttt{(}\forall\texttt{x1)(}\forall\texttt{x2)((L1(}x\texttt{)}\wedge\texttt{L1(x1)}\wedge\texttt{L1(x2)}\wedge\texttt{ben(}x\texttt{,x1)} \\ \wedge\texttt{ben(}x\texttt{,x2))}\to\texttt{x1=x2)} \tag{27.9}$$

Diese Formel ist schematisch, weil sie eine schematische Variable enthält, nämlich

x. Je nach Wahl für x fällt die Formel anders aus. Wählen wir für x zum Beispiel /`x17`/, so bekommen wir

$$(27.10) \qquad \varepsilon_1(\mathtt{x17}) := \mathtt{(\forall x1)(\forall x2)((L1(x17)\wedge L1(x1)} \\ \mathtt{\wedge L1(x2)\wedge ben(x17,x1)\wedge ben(x17,x2))\to x1=x2)}$$

Wie wir im vorigen Kapitel gesehen haben, charakterisiert diese Formel die Eigenschaft „`x17` ist ein Endbahnhof". Dies gilt allerdings nur mit kleinen Einschränkungen. So bekämen wir zum Beispiel, wenn wir alle Vorkommen von x durch /`x1`/ ersetzten die Formel

$$(27.11) \qquad \varepsilon_1(\mathtt{x1}) := \mathtt{(\forall x1)(\forall x2)((L1(x1)\wedge L1(x1)\wedge L1(x2)\wedge ben(x1,x1)} \\ \mathtt{\wedge ben(x1,x2))\to x1=x2)}$$

Diese Formel ist aber in allen Liniennetzen von allen Bahnhöfen erfüllt! Denn /`ben(x1,x1)`/ ist immer falsch, da die Nachbarschaftsrelation irreflexiv ist. Was wir hier haben ist, dass die neue Formel nicht dieselbe Menge definiert im Sinne von Definition 26.4. Problematisch ist außer der Einsetzung von /`x1`/ auch die Einsetzung von /`x2`/, also einer weiteren Variable, die wir in ζ_1 brauchen. Die Formel ζ_1 sagt, dass es für die Linie 1 genau zwei Endhaltestellen gibt.

$$(27.12) \qquad \zeta_1 := \mathtt{(\exists x0)(\exists x1)(\forall x2)(}\varepsilon_1\mathtt{(x0)\wedge}\varepsilon_1\mathtt{(x1)\wedge(\neg x0=x1)\wedge}\varepsilon_1\mathtt{(x2)} \\ \mathtt{\to(x0=x1\vee x0=x2\vee x1=x2))}$$

Was also ist zu tun?

Zunächst einmal wollen wir das Problem analysieren. Offenkundig ist es nicht in Ordnung, solche Variable einzusetzen, die in der Formel bereits gebunden vorkommen. Denn der Quantor verlangt, dass wir alle Varianten für die Belegung /`x1`/ ausprobieren, also darf /`x1`/ nicht unter den Variablen sein, die wir für x einsetzen wollen. Man hat sich für diesen Fall einen Trick ausgedacht, nämlich die sogenannte *gebundene Umbenennung*. Bei der Umbenennung nutzt man aus, dass die Wahl der Variablen eigentlich keine große Rolle spielt, sofern sie gebunden ist. Anstelle der problematischen Variable hätte auch eine andere stehen können. Ich erinnere an die Notation fv(φ) für die Menge der freien Variablen von φ. Wir werden hier noch genauer hinschauen und nicht einfach nur katalogisieren, welche Variablen frei sind sonder zusätzlich, wo sie frei vorkommen und wo nicht.

Definition 27.1 *Ein* ***Quantor*** *ist eine Zeichenkette der Form* $(\forall x)$ *oder* $(\exists x)$*. Es sei* φ *eine Formel,* x *eine Variable. Ein Vorkommen von* x *in* φ *heißt* ***uneigentlich****, falls es in einem Quantor vorkommt. Ein (eigentliches) Vorkommen von* x *in* φ *heißt* ***gebunden****, falls es in einer einer Teilformel der Form* $(\forall x)\eta$ *oder* $(\exists x)\eta$ *enthalten ist. In dieser Formel ist* η *der* ***Skopus*** *des Quantors. Wir sagen, das Vorkommen des Quantors* $(\exists x)$ *bzw.* $(\forall x)$ ***bindet*** *ein Vorkommen von x, falls der Skopus dieses Quantorenvorkommens der kleinste aller Quantorenvorkommen mit* x *als uneigentlicher Variable ist, die das Vorkommen der Variablen* x *enthalten. Ist ein Vorkommen nicht gebunden, so heißt es* ***frei****.*

Normalerweise zählt man nur eigentliche Vorkommen.

Achtung: Ein Quantor kann nur eigentliche Vorkommen der in ihm uneigentlich auftretenden Variable binden; alle anderen Variablen können durch ihn in keinem ihrer Vorkommen gebunden sein. Der Quantor kann eine Variable in seinem Skopus aber nur dann binden, wenn kein Quantor dazwischentritt, der dieselbe Variable uneigentlich enthält. Man kennt diese Bedingung in der Sprachwissenschaft als Interventionseffekt. In einer Formel $(\forall x)(\cdots(\exists x)(\cdots x \cdots)\cdots)$ wird das Vorkommen von x durch den Existenzquantor gebunden, obwohl es auch im Skopus des Allquantors ist. Dagegen ist der Allquantor der Binder in $(\forall x)(\cdots(\exists y)(\cdots x \cdots)\cdots)$, wenn $y \neq x$. Zu jedem gebundenen Vorkommen einer Variablen gibt es einen eindeutiges Quantorenvorkommen, das dieses Variablenvorkommen bindet.

Da eine Variable mehrfach auftritt, muss man jedes Vorkommen extra anschauen. Betrachten wir folgende Formel

(27.13) $\texttt{ben(x0,}\underline{\texttt{x1}}\texttt{)}\wedge\texttt{(}\exists\underline{\texttt{x1}}\texttt{)(L1(}\underline{\texttt{x1}}\texttt{)}\wedge\texttt{ben(}\underline{\texttt{x1}}\texttt{,x0))}$

Wir haben hier vier Vorkommen. Das erste ist frei, weil es keine Formel mit Quantor gibt, die es enthält. Von den anderen drei sind das erste uneigentlich, die anderen zwei gebunden; in jedem Fall ist es dieselbe Formel, nämlich

(27.14) $\texttt{(}\exists\underline{\texttt{x1}}\texttt{)L1(}\underline{\texttt{x1}}\texttt{)}\wedge\texttt{ben(}\underline{\texttt{x1}}\texttt{,x0))}$

Bei der Ersetzung müssen wir nun freie und gebundene Vorkommen verschieden behandeln. Ich erinnere an das Koinzidenzlemma. Es besagt, dass der Wahrheitswert nur auf eine Änderung der Werte der freien Variablen reagiert.

Definition 27.2 *Sei* y *eine Variable, die in* φ *nicht vorkommt. Es bezeichnet dann* $[y/x]\varphi$ *die Ersetzung von allen freien Vorkommen von* x *in* φ *durch* y*.*

Es gilt nun, wie man leicht sieht, Folgendes.

Lemma 27.3 *Es sei φ eine Formel, $\mathfrak{M}$ eine Struktur. Ist dann y eine Variable, die in φ nicht frei vorkommt, und ist $\beta(y) = \beta(x)$, so ist $\langle\mathfrak{M},\beta\rangle \vDash [y/x]\varphi$ gdw. $\langle\mathfrak{M},\beta\rangle \vDash \varphi$.*

Die Bedingung, dass y nicht frei in φ vorkommt, ist hier eigentlich nicht nötig, da wir ohnehin $\beta(y) = \beta(x)$ fordern. Wichtig ist aber, dass nur die *freien* Vorkommen ersetzt werden.

Lemma 27.4 (Gebundene Umbenennung) *Es sei $(\forall x)\varphi$ eine Formel und y eine Variable, die nicht in φ auftritt. Dann gilt für alle Strukturen $\mathfrak{M}$ und Belegungen β: $\langle\mathfrak{M},\beta\rangle \vDash (\forall x)\varphi$ gdw. $\langle\mathfrak{M},\beta\rangle \vDash (\forall y)([y/x]\varphi)$. Analog für eine Formel der Form $(\exists x)\varphi$.*

Beweis. Es sei $\langle\mathfrak{M},\beta\rangle \vDash (\forall x)\varphi$. Wir haben zu zeigen, dass $\langle\mathfrak{M},\beta\rangle \vDash (\forall y)([y/x]\varphi)$. Dazu wählen wir eine y-Variante β' von β. Wir werden zeigen, dass $\langle\mathfrak{M},\beta'\rangle \vDash [y/x]\varphi$. Dazu sei β'' eine x-Variante von β' mit $\beta''(x) = \beta'(y)$. Nach dem Koinzidenlemma ist dann $\langle\mathfrak{M},\beta'\rangle \vDash [y/x]\varphi$ genau dann der Fall, wenn $\langle\mathfrak{M},\beta''\rangle \vDash [y/x]\varphi$. Nun ist $\beta''(x) = \beta'(y)$, also ist dies soviel wie $\langle\mathfrak{M},\beta''\rangle \vDash \varphi$. Wir wählen nun eine y-Variante β''' von β'' mit $\beta'''(y) := \beta(y)$. Es ist β''' eine x-Variante von β, mithin ist $\langle\mathfrak{M},\beta'''\rangle \vDash \varphi$. Nach dem Koinzidenzlemma ist dann $\langle\mathfrak{M},\beta''\rangle \vDash \varphi$, was zu zeigen war. ⊣

Dazu noch ein Stück Notation. Wir schreiben $\varphi \equiv \chi$, falls für alle Strukturen $\mathfrak{M}$ und für alle Belegungen β gilt $\langle\mathfrak{M},\beta\rangle \vDash \varphi$ gdw. $\langle\mathfrak{M},\beta\rangle \vDash \chi$. Wir sagen auch, die Formeln φ und χ seien **semantisch äquivalent**. Dies ist gleichbedeutend damit, dass die folgende Formel eine Tautologie ist:

$$(\forall x_1)(\forall x_2)\cdots(\forall x_n)((\varphi\to\chi)\wedge(\chi\to\varphi)) \tag{27.15}$$

Hierbei sind $x_1, \cdots, x_n$ die in φ oder χ frei auftretenden Variablen.

Nun betrachten wir noch einmal die Ersetzung von $\varepsilon_1(x)$. Wenn es nun nicht möglich ist, /`x1`/ für x einzusetzen, dann tut man Folgendes: Die Variable /`x1`/ ist ja in $\varepsilon_1(x)$ gebunden. Nun dürfen wir sie nach Satz 27.4 umbenennen. Wir wählen jetzt eine Variable aus, die nicht in der Formel vorkommt und von /`x1`/ verschieden ist, etwa /`x24`/. Wir ersetzen jetzt alle gebundenen Vorkommen von /`x1`/ durch /`x24`/. Das ergibt die Formel

$$(\forall \mathtt{x24})(\forall \mathtt{x2})((\mathtt{L1}(x)\wedge\mathtt{L1}(\mathtt{x24})\wedge\mathtt{L1}(\mathtt{x2})\wedge\mathtt{ben}(x,\mathtt{x24}) \wedge\mathtt{ben}(x,\mathtt{x2}))\to\mathtt{x24}=\mathtt{x2}) \tag{27.16}$$

Anschließend setzen wir /x1/ für x ein:

(27.17) $$\texttt{(∀x24)(∀x2)((L1(x1)∧L1(x24)∧L1(x2)∧ben(x1,x24)∧ben(x1,x2))→x24=x2)}$$

Dies ist die Formel, die wir in ζ_1 (siehe (27.18)) einsetzen werden. Ebenso suchen wir für die Einsetzung von /x2/ ein passende Variable (hier /x17/). Damit wird jetzt endlich ζ_1 eine konkrete Zeichenkette:

(27.18)
$$\begin{aligned}
\zeta_1 &= \texttt{(∀x0)(∀x1)(∀x2)(}\varepsilon_1\texttt{(x0)∧}\varepsilon_1\texttt{(x1)∧}\varepsilon_1\texttt{(x2)}\\
&\qquad \texttt{→(x0=x1∨x0=x2∨x1=x2))}\\
&= \texttt{(∀x0)(∀x1)(∀x2)((∀x1)(∀x2)((L1(x0)}\\
&\qquad \texttt{∧L1(x1)∧L1(x2)∧ben(x0,x1)∧ben(x0,x2))}\\
&\qquad \texttt{→x1=x2)∧(∀x24)(∀x2)((L1(x1)∧L1(x24)}\\
&\qquad \texttt{∧L1(x2)∧ben(x1,x24)∧ben(x1,x2))→x24=x2)}\\
&\qquad \texttt{∧(∀x1)(∀x17)((L1(x1)∧L1(x2)∧L1(x17)}\\
&\qquad \texttt{∧ben(x2,x1)∧ben(x2,x17))→x17=x2)}\\
&\qquad \texttt{→(x0=x1∨x0=x2∨x1=x2))}
\end{aligned}$$

(Wir haben natürlich gemäß Klammerkonvention jede Menge Klammern gespart. Ich könnte diese einsetzen, aber dann wäre die Formel komplett unlesbar.) Es ist klar, dass wir eine ganze Menge anderer Variablen anstelle von /x24/ hätten wählen können. Das Verfahren ist also unbestimmt (obwohl wir eine eindeutige Wahl durchaus organisieren können, indem wir immer verlangen, dass man die Variable mit der kleinstmöglichen Zahl als Index nimmt). Rein semantisch gesehen, also bis auf semantische Äquivalenz ≡, ist aber die Wahl unerheblich. Deswegen müht man sich hier auch nicht um Eindeutigkeit.

Übungen

[99] Es sei folgende Formel gegeben.

$$\delta(y) := \texttt{(∀x2)(}y\texttt{<x2→(∃x1)(}y\texttt{<x1∧x2<x1+1))}$$

Setzen Sie für y jeweils die Variablen x0, x1 bzw. x2 ein (unabhängig voneinander, nicht hintereinander einsetzen!) und schreiben Sie das Ergebnis auf.

[100] Sei $\varphi(x)$ eine Formel mit der einzigen freien Variable x, die in der Struktur eine Menge A definiert. Mit der oben definierten Substitution gilt nun Folgendes: Die Formel $[y/x]\varphi(x)$ definiert ebenfalls die Menge A. Verifizieren Sie dies für die Struktur des Liniennetzes anhand der Formel $\varepsilon_1(x)$.

Kapitel 28

Ein Kalkül

Den Kalkül aus Kapitel 21 werde ich nun zu einem Kalkül für Prädikatenlogik umarbeiten. Wiederum werde ich einen Unterschied zwischen einer H-Ableitung und einer C-Ableitung machen.

Definition 28.1 *Es sei Δ eine Menge von Formeln und φ eine Formel. Wir schreiben $\Delta \vDash \varphi$, falls für alle Strukturen $\mathfrak{M}$ und alle Belegungen β gilt: wenn $\langle \mathfrak{M}, \beta \rangle \vDash \delta$ für alle $\delta \in \Delta$, so ist auch $\langle \mathfrak{M}, \beta \rangle \vDash \varphi$.*

Ist insbesondere $\Delta = \varnothing$, so bedeutet $\Delta \vDash \varphi$ nichts anderes, als dass φ stets gültig ist, dh $\mathfrak{M} \vDash \varphi$. Man beachte nämlich, dass $\mathfrak{M} \vDash \chi$ bedeutet, dass für alle Belegungen β: $\langle \mathfrak{M}, \beta \rangle \vDash \chi$. So ist zum Beispiel $\vDash$ `(S(x0)→S(x0))`. Denn in jeder Struktur $\mathfrak{M}$ und für jede Belegung ist $\langle \mathfrak{M}, \beta \rangle \vDash$ `(S(x0)→S(x0))`.

Ist aber χ = `ben(x0,x1)→ben(x1,x0)`, so bedeutet $\mathfrak{M} \vDash \chi$ nichts anderes, als dass I(`ben`) symmetrisch ist. Denn sei β(`x0`) $= a$ und β(`x1`) $= b$. Dann ist $\langle \mathfrak{M}, \beta \rangle \vDash \chi$ genau dann, wenn aus $\langle a, b \rangle \in I$(`ben`) folgt $\langle b, a \rangle \in I$(`ben`). Dies gilt nun ganz allgemein.

Lemma 28.2 *Falls $\mathfrak{M} \vDash \chi$, so auch $\mathfrak{M} \vDash (\forall x)\chi$.*

Beweis. Es sei $\mathfrak{M} \vDash \chi$. Sei β eine Belegung. Ist dann β' eine x-Variante von β, so gilt auch $\langle \mathfrak{M}, \beta' \rangle \vDash \chi$. Also ist $\langle \mathfrak{M}, \beta \rangle \vDash (\forall x)\chi$. ⊣

Daraus können wir folgenden Schluss ziehen.

Lemma 28.3 *Falls $\Delta \vdash' \varphi$, so auch $\Delta \vdash' (\forall x)\varphi$.*

Beweis. Sei $\mathfrak{M}$ beliebig mit $\mathfrak{M} \vDash \delta$ für alle $\delta \in \Delta$. Nach Voraussetzung ist dann $\mathfrak{M} \vDash \varphi$, und nach dem obigen Lemma auch $\mathfrak{M} \vDash (\forall x)\varphi$. ⊣

Dies sind die Schlussregeln für Quantoren. Zusätzlich haben wir auch noch die alten Schlussregeln für Aussagen. Diese wurden in (21.1) festgelegt. Da wir nun aber keine Variable für Aussagen zur Verfügung haben, müssen wir sie diesmal als schematische Regeln auffassen.

(28.1)

(pb)	$\phi \to \chi \to \phi$
(fd)	$(\phi \to \chi \to \psi) \to (\phi \to \chi) \to \phi \to \psi$
(pc)	$((\phi \to \chi) \to \phi) \to \phi$
(l∧)	$\phi \to \chi \to \phi \wedge \chi$
(r∧)	$\phi \to \chi \to \chi \wedge \phi$
(∧l)	$\phi \wedge \chi \to \phi$
(∧r)	$\phi \wedge \chi \to \chi$
(l∨)	$(\phi \to \psi) \to (\chi \to \psi) \to (\phi \vee \chi) \to \psi$
(r∨)	$(\phi \to \psi) \to (\chi \to \psi) \to (\chi \vee \phi) \to \psi$
(∨l)	$\phi \to (\phi \vee \chi)$
(∨l)	$\chi \to (\phi \vee \chi)$
(e¬)	$(\phi \to \mathtt{f}) \to \neg \phi$
(¬e)	$\neg \phi \to \phi \to \mathtt{f}$
(dn)	$\neg\neg \phi \to \phi$
(efq)	$\mathtt{f} \to \phi$
(t)	$\mathtt{w}$
(∃d1)	$(\exists x)\phi \to \neg(\forall x)\neg \phi$
(∃d2)	$\neg(\forall x)\neg\phi \to (\exists x)\phi$
(∀e)	$(\forall x)\phi \to [t/x]\varphi$
(∀d)	$(\forall x)(\phi \to \chi) \to (\forall x)\phi \to (\forall x)\psi$

Für diesen neuen Kalkül müssen wir den Begriff der Ableitung leicht modifizieren.

Definition 28.4 *Eine **H-Ableitung** von φ ist eine Folge $\langle \delta_i : i < n+1 \rangle$ derart, dass $\delta_n = \varphi$ und für jedes $i < n+1$ gilt:*

① *δ_i ist eine Einsetzungsinstanz eines Axioms, wobei in (∀e) nur solche t gewählt werden dürfen, deren Variable nicht frei in ϕ auftreten, oder*

② *$\delta_i = (\forall x)\phi$, und es existiert ein $j < i$ mit $\delta_j = [y/x]\phi$, wo y nicht frei in ϕ vorkommt, oder*

③ *es existieren $j, k < i$ derart, dass $\delta_k = \delta_j \to \delta_i$ ist.*

Wir sagen, φ sei ***H-ableitbar****, in Zeichen $\vdash \varphi$, wenn es eine H-Ableitung von φ gibt.*

Satz 28.5 *Genau dann ist φ H-ableitbar, wenn $\vDash \varphi$.*

Wir erweitern dies sofort zum C-Kalkül.

Definition 28.6 *Eine* ***C-Ableitung*** *von φ aus Σ ist eine Folge $\langle \delta_i : i < n+1 \rangle$ derart, dass $\delta_n = \varphi$ und für jedes $i < n+1$ gilt:*

① $\delta_i \in \Sigma$, *oder*

② δ_i *ist eine Einsetzungsinstanz eines Axioms, wobei in* $(\forall e)$ *und* $(\exists e)$ *nur solche t gewählt werden dürfen, deren Variable nicht frei in ϕ auftreten, oder*

③ $\delta_i = (\forall x)\phi$, *und es existiert ein $j < i$ mit $\delta_j = [y/x]\phi$, wo y nicht frei in δ_k für irgendein $k < i$, oder*

④ *es existieren $j, k < i$ derart, dass $\delta_k = \delta_j \to \delta_k$ ist.*

⑤

Wir sagen, φ sei aus Σ ***C-ableitbar****, in Zeichen $\Sigma \vdash \varphi$, wenn es eine C-Ableitung von φ aus Σ gibt.*

Es ist nun klar, dass wenn φ im H-Kalkül ableitbar ist, so ist φ aus $\varnothing$ C-herleitbar. Ist auf der anderen Seite φ aus $\varnothing$ C-herleitbar, so ist φ auch H-herleitbar. Insofern fallen die Begriffe „C-ableitbar" und „H-ableitbar" zusammen und wir sprechen fortan nur von „ableitbar". Wir können im Übrigen für den C-Kalkül auch das Deduktionstheorem beweisen.

Satz 28.7 *Genau dann ist φ aus $\Delta; \chi$ C-ableitbar, wenn $\chi \to \varphi$ aus Δ C-ableitbar ist.*

Satz 28.8 (Vollständigkeit) *Genau dann ist $\Sigma \vdash \varphi$, wenn auch $\Sigma \vDash \varphi$.*

Übungen

[101] Leiten Sie folgende Formeln ab.

$$(\exists \mathtt{x0})(\phi\wedge\chi)\to(\exists \mathtt{x0})\phi\wedge(\exists \mathtt{x0})\chi \quad (\exists \mathtt{x0})\phi\wedge(\exists\chi)\to(\exists \mathtt{x0})(\phi\wedge\chi).$$

[102] Zeigen Sie, dass $((\forall \mathtt{x0})\phi\to(\forall \mathtt{x0})\psi)\to(\forall \mathtt{x0})(\phi\to\chi)$ nicht allgemein gilt. *Hinweis.* Nehmen Sie für ϕ eine Eigenschaft von natürlichen Zahlen, die nicht auf alle Zahlen zutrifft, etwa $\mathtt{x0}=\mathtt{0}$. Jetzt wählen Sie für ψ eine Eigenschaft von natürlichen Zahlen, sodass $\mathtt{x0}=\mathtt{0}\to\psi$ nicht allgemein richtig ist.

Kapitel 29

Syllogistik

Die moderne Logik, die ich bis hierher skizziert habe, ist nicht die einzige Art Logik. In der europäischen Tradition war bis in das 19. Jahrhundert die vorherrschende Logik die sogenannte Syllogistik. Für eine ausührliche Darstellung verweise ich auf Parsons 2014. Die Syllogistik geht auf Aristoteles zurück, vor allem die Schriften, die um 340 v.Chr. erschienen (De Intepretatione, Analytica Priora, Analytica Posteriora). Sie wurde vor allem im Mittelalter vorangetrieben, bis dann mit der Renaissance die Logik an Bedeutung verlor. Das hat unter anderem damit zu tun gehabt, dass Logik sehr eng mit Theologie verbunden war. Die großen Gelehrten des Mittelalters, William von Ockham (1288 - 1347), Petrus Abaelardus (1079 - 1142), Duns Scotus (1266 - 1308), Peter von Spanien (13. Jhdt.), hatten sich immer auch zu theologischen Fragen geäußert. Ich gebe hier folgendes Beispiel. Man anerkannte die Transitivität der Identität (wenn a b ist und a c ist, so ist a b), aber die Trinitätslehre machte Schwierigkeiten: „Der Vater ist Gott." und „Der Sohn ist Gott." wurden für wahr befunden, aber „Der Vater ist der Sohn." war häretisch. John Buridan schlug deswegen z.B. vor, der Schluss gelte für alle Dinge außer den göttlichen. Andere Denker gingen allerdings subtiler vor.

Als im Mittelalter neue Schriften des Aristoteles auftauchten, die mit der vorherrschenden Lehrmeinung im Konflikt waren, verbot der Bischof von Paris, Etienne Tempier, 1270 und 1277 kurzerhand gewisse Lehrmeinungen. Uns sind allerdings Vorlesungen von Duns Scotus erhalten, in denen er die vom Bischof hochgehaltenen Lehren eine nach dem anderen verwirft, und zwar indem er rein logisch argumentiert.

In der Renaissance hatte man für solche Dinge nicht mehr viel übrig. Zudem begann man sich für die Naturwissenschaften zu interessieren. Praktische Erfolge zählten mehr als die Frage, wie beschaffen Gott sei. Leider ging mit der Zeit auch das Verständnis für Logik verloren, und nur noch wenige versuchten sich daran (unter anderem Gottfried Wilhelm Leibniz (1646 - 1716)). Das änderte sich erst im 19. Jahrhundert, als George Boole (1815 - 1864), Augustus de Morgan (1806 - 1871) und Gottlob Frege (1848 - 1925) sich daran machten, die Gesetze der Logik zu untersuchen. Dabei verdanken wir vor allem Frege eine völlig neue Konzeption der Logik, insbesondere der Prädikatenlogik, die aus der sogenannten *Begriffsschrift* hervorging.

Trotz der Tatsache, dass diese neuen Entwicklungen einen großen Fortschritt darstellten, ist die Beschäftigung mit der Syllogistik lohnenswert, insbesondere weil sie näher an der natürlichen Sprache ist. Sie ist nämlich zunächst einmal innerhalb der natürlichen Sprache verfasst. Sie arbeitet anders als die moderne Logik nicht mit einer völlig neuen Syntax, sondern benutzt die natürliche Sprache selbst zur Darstellung der Inhalte. Das birgt Risiken, wie wir wissen. Die Vorteile sind aber ebenso klar: Wir können lernen, wie wir innerhalb der Sprache selbst argumentieren können, ohne erst einmal eine neue Sprache lernen zu müssen.

Ich gebe ein Beispiel. Eine Schlussregel hieß *Modus*. Modi bekamen Namen, die ich später noch erläutere. Einer dieser Modi war der Modus Barbara. Dieser geht so.

$$
\text{(29.1)} \qquad \begin{array}{ll} & \texttt{Alle } B \texttt{ sind } C\texttt{.} \\ & \texttt{Alle } A \texttt{ sind } B\texttt{.} \\ \hline \therefore & \texttt{Alle } A \texttt{ sind } C\texttt{.} \end{array}
$$

Um diesen Modus anzuwenden, müssen wir für *A*, *B* und *C* lediglich geeignete NPs im Plural einsetzen.

$$
\text{(29.2)} \qquad \begin{array}{ll} & \texttt{Alle Ameisen sind Insekten.} \\ & \texttt{Alle Insekten sind Lebewesen.} \\ \hline \therefore & \texttt{Alle Ameisen sind Lebewesen.} \end{array}
$$

Dass dieser Schluss korrekt ist, sieht man unmittelbar ein. Ebenso kann man sich sehr schnell davon überzeugen, dass der Modus Barbara im Allgemeinen korrekt ist. Ein weiterer Typ Sätze waren solche, die Eigennamen enthielten. Diese hatten besonderen Status.

In der Syllogistik gibt es vier Sorten von Sätzen:

1. `Alle Esel sind Tiere.`
2. `Kein Esel ist eine Katze.`
3. `Einige Esel sind Tiere.`

4. `Einige Esel sind keine Katzen.`

Ich erwähne hier die Genauigkeit wegen, dass der Gebrauch des Plurals nicht bedeutet, dass notwendig eine Mehrzahl von Eseln gemeint ist. Es ist lediglich so, dass die Singularform im Deutschen eher generisch interpretiert wird. /`Ein Esel ist ein Tier.`/ bedeutet so viel wie /`Alle Esel sind Tiere.`/; in der Bedeutung, dass es einen Esel gibt, der auch Tier ist, klingt es eher künstlich. Im Lateinischen war das übrigens ebenso. Da die Texte im Mittelalter sämtlich in Latein verfasst wurden und es im Lateinischen keinen Artikel gab, klangen diese Beispiel damals anders. Ich zitiere Beispiele aus Boethius, welches eine der Hauptquellen für die antike Syllogistik war.

1. `Omnis homo sapiens est.` (Jeder Mensch ist weise.)
2. `Nullus homo sapiens ist.` (Kein Mensch ist weise.)
3. `Quidam homo sapiens est.` (Ein gewisser Mensch ist weise.)
4. `Quidam homo sapiens non est.` (Ein gewisser Mensch ist nicht weise.)
5. `Socrates sapiens est.` (Sokrates ist weise.)

Die ersten vier Sätze betreffen Subjekte mit allgemeinen Termini. Ich liste diese Typen gesondert auf und gebe auch gleich ihre Entsprechungen in der modernen Prädikatenlogik.

1. Affirmative Allsätze: Sie beginnen mit /`alle`/, Kürzel A.
   Prädikatenlogisch: $((\exists x)\varphi)\wedge(\forall x)(\varphi\rightarrow\chi)$;

2. Negative Allsätze: Sie beginnen mit /`kein(e)`/, Kürzel E.
   Prädikatenlogisch: $(\forall x)(\varphi\rightarrow\neg\chi)$;

3. Affirmative Existenzsätze: Sie beginnen mit /`manche(r)`/ oder /`einige`/, Kürzel I.
   Prädikatenlogisch: $(\exists x)(\varphi\wedge\chi)$; und

4. Negative Existenzsätze: Sie haben die Form /`manche`$\cdots$ `sind kein(e)`/, oder /`manche(r)`$\cdots$ `sind nicht`/, Kürzel O.
   Prädikatenlogisch: $((\forall x)\neg\varphi)\vee(\exists x)(\varphi\wedge\neg\chi)$.

Ein Modus besteht aus drei Sätzen: eine sogenannte große Prämisse (auch **Obersatz** genannt), eine kleine Prämisse (oder **Untersatz**) und einem Schluss. Die große Prämisse ist lediglich diejenige, die zuerst genannt wird. Hier ist zum Beispiel der Modus Celarent.

$$\begin{array}{ll} & \texttt{Kein } A \texttt{ ist } B\texttt{.} \\ & \texttt{Alle } C \texttt{ sind } A\texttt{.} \\ \hline \therefore & \texttt{Kein } C \texttt{ ist } B\texttt{.} \end{array} \tag{29.3}$$

In moderne Logik übersetzt sieht dies so aus:

$$\begin{array}{ll} & (\forall x)(\varphi(x)\to(\neg\chi(x))) \\ & ((\exists x)\psi(x))\wedge(\forall x)(\psi(x)\to\varphi(x)) \\ \hline \therefore & (\forall x)(\psi(x)\to(\neg\chi(x))) \end{array} \tag{29.4}$$

Hierbei sind φ, ψ und χ Formeln der Logik, während A, B und C sprachliche Ausdrücke sind. Unter Weglassung an sich unnötiger Details sieht das Ganze so aus.

$$\begin{array}{ll} & (\forall x)(\varphi\to\neg\chi) \\ & ((\exists x)\psi)\wedge(\forall x)(\psi\to\varphi) \\ \hline \therefore & (\forall x)(\psi\to\neg\chi) \end{array} \tag{29.5}$$

Die Gültigkeit ist leicht zu sehen. Gäbe es ein a, das sowohl ψ als auch χ wäre, so wäre dieses a auch ein φ (Untersatz), und damit ein φ, das auch ein χ ist, im Widerspruch zum Obersatz.

Auf der anderen Seite ist für den modernen Logiker Celarent nichts weiter als eine besondere Form von Modus Barbara (mit negativem Prädikat).

Im Unterschied zur modernen Logik bedeutet in der Syllogistik der Satz /`Alle` A `sind` B`.`/ unter anderem, dass es mindesten ein A gibt. Gibt es nicht mindestens ein A, so ist der Satz falsch. Man spricht hier von einer *Existenzpräsupposition*. In dem Fall, wo es keine Katzen gibt, ist sowohl der Satz /`Alle Katzen sind grau.`/ wie auch der Satz /`Alle Katzen sind nicht grau.`/ nicht wahr. Deswegen ist in der Syllogistik der Modus Darapti gültig:

$$\begin{array}{ll} & \texttt{Alle } B \texttt{ sind } C\texttt{.} \\ & \texttt{Alle } A \texttt{ sind } B\texttt{.} \\ \hline \therefore & \texttt{Einige } A \texttt{ sind } C\texttt{.} \end{array} \tag{29.6}$$

Dies folgt mittels Modus Barbara und der einfachen Tatsache, dass wir aus /`Alle` A `sind` C`.`/ auch folgern dürfen /`Einige` A `sind` C`.`/. In der Syllogistik braucht man aber immer zwei Prämissen, und so ist dieser letzte Schluss, obwohl richtig, keine Schlussfigur der Syllogistik.

Die Existenzpräsupposition ist nicht besser oder schlechter als die Semantik des Quantors in der Prädikatenlogik. Sie folgt der allgemeinen Intuition der Sprecher, dass ein Allsatz nur dann Sinn macht, wenn es Gegenstände gibt, über die er redet. Die negativen All- und Existenzsätze tragen diese Existenzpräsupposition jedoch *nicht*. Es ist also /`Einige` A `sind nicht` B`.`/ auch dann wahr, wenn es keine A gibt. Das liegt daran, dass /`nicht`/ hier als Satznegation gelesen wurde. Etwas anderes wäre im Deutschen /`Einige` A `sind keine` B`.`/, wo die Negation nicht den Satz betrifft sondern das Prädikat. (Man bedenke allerdings, dass die Abhandlungen im Mittelalter alle im Lateinischen verfasst wurden, welches diesen Unterschied nicht kannte.) Dieser Satz ist existentiell und affirmativ, behauptet also das Vorhandensein von A generell, und insbesondere das eines A, das nicht B ist. Dabei war auch erlaubt, Eigennamen zu negieren, etwa in /`Einige Griechen sind nicht Sokrates.`/. Das Prädikat /`nicht Sokrates`/ konnte natürlich kein Name sein, es war also technisch infinit, wie schon Aristoteles bemerkte. Die Negation des Terms wurde deswegen delimitierend genannt. Schon in Mittealter begann man, die Satznegation von der delimitierenden Negation zu trennen und mit Schlussformen zu experimentieren, die auch delimitierende Negation einschloss.

Die Namen der Modi sind dreisilbige Worte, bei denen die Silbenvokale den Typ des Obersatzes (1. Silbe), des Untersatzes (2. Silbe) und des Schlusses (3. Silbe) bezeichnet. Bei dem Modus Barbara sind folglich alle drei Sätze positive Allsätze, bei Celarent sind Obersatz und Schluss negative Allsätze, der Untersatz ein positiver Allsatz. Entsprechend ist beim Modus Festino der Obersatz ein negativer Allsatz, der Untersatz ein positive Existenzsatz, und der Schluss ein negativer Allsatz. Hier ist der Modus in seiner allgemeinen Form.

$$\begin{array}{rl} & \texttt{Kein } A \texttt{ ist } B\texttt{.} \\ & \texttt{Manche } C \texttt{ sind } B\texttt{.} \\ \hline \therefore & \texttt{Manche } C \texttt{ sind keine } A\texttt{.} \end{array} \tag{29.7}$$

Hier ist ein Beispiel.

$$\begin{array}{rl} & \texttt{Keine Quadratzahl ist eine Primzahl.} \\ & \texttt{Manche Fibonaccizahlen sind Primzahlen.} \\ \hline \therefore & \texttt{Manche Fibonaccizahlen sind keine Quadratzahlen.} \end{array} \tag{29.8}$$

Die Gültigkeit von Modi lässt sich außer mit der Reduktion auf Prädikatenlogik auch mit Hilfe von sogenannten Venn-Diagrammen prüfen. Diese sind identisch mit den Mengendiagrammen aus Formale Methoden 1. Es sind immer drei Mengen beteiligt, A, B und C. Die Übersetzung der Satztypen ist wie folgt:

1. „Alle A sind B." $\rightsquigarrow A \subseteq B$.
2. „Kein A ist B." $\rightsquigarrow A \cap B = \varnothing$.
3. „Ein A ist B." $\rightsquigarrow A \cap B \neq \varnothing$.
4. „Ein A ist nicht B." $\rightsquigarrow A \subsetneq B$.

Modus Barbara, so übersetzt, schließt von $A \subseteq B$ und $B \subseteq C$ auf $A \subseteq C$.

Schon Aristoteles bemühte sich, die Syllogismen zu beweisen. Dabei benutzte er gewisse Schlussverfahren. Eines davon nennt er ekthesis, wörtlich *Herausstellung*. Dies ist ähnlich zu der existentiellen Instantiierung. Dies bestand darin, dass man aus dem Satz /`Ein` A `ist` B`.`/ sowohl /n `ist` A`.`/ als auch /n `ist` B`.`/ ableiten durfte. Dabei muss n ein frischer Name sein, insbesonde durfte er noch nicht in der Ableitung vorgekommen sein. Man durfte also nicht ableiten /`Sokrates ist` A`.`/ oder /`Sokrates ist` B`.`/. Dann könnten wir ja aus /`Einige Tiere sind Katzen.`/ schließen, dass sowohl /`Sokrates ist ein Tier`./ wie auch /`Sokrates ist eine Katze.`/ wahr sind.

Die Umkehrung der Ekthesis ist übrigens auch richtig: Hat man /n `ist` A`.`/ und /n `ist` B`.`/, so darf man /`Ein` A `ist` B`.`/ schließen. Diesmal aber gibt es keine Bedingung an n. Aus /`Sokrates ist ein Philosoph.`/ und /`Sokrates ist ein Grieche.`/ dürfen wir jetzt schließen

(29.9) `/Einige Griechen sind Philosophen./`

Ein anderes Schlussverfahren war die Reduktion. Konnte man aus den Prämissen zwei sich ausschließende Sätze ableiten, so war das Gegenteil einer der Prämissen aus den anderen Prämissen bewiesen. Genauer: Kann man zwei konträre Propositionen ableiten, so ist eine der Prämissen durch eine dazu kontradiktorische Prämisse zu ersetzen (siehe unten für die Begriffe).

Ein anderes bekanntes Schlussverfahren war die Konversion.

$$(29.10) \qquad \frac{\texttt{Einige } A \texttt{ sind } B\texttt{.}}{\therefore\ \texttt{Einige } B \texttt{ sind } A\texttt{.}} \qquad \frac{\texttt{Kein } A \texttt{ ist } B\texttt{.}}{\therefore\ \texttt{Kein } B \texttt{ ist } A\texttt{.}}$$

Ein weiteres war die Konversion *per accidens*, die wir oben schon verwendet haben.

$$(29.11) \qquad \frac{\texttt{Alle } A \texttt{ sind } B\texttt{.}}{\therefore\ \texttt{Manche } A \texttt{ sind } B\texttt{.}} \qquad \frac{\texttt{Kein } A \texttt{ ist } B\texttt{.}}{\therefore\ \texttt{Manche } A \texttt{ sind nicht } B\texttt{.}}$$

Man beachte, dass bei die zweite Schlussregel deswegen richtig ist, weil der negative Existenzsatz keine Existenzpräsupposition trägt.

Besonders bekannt ist die aristotelische Logik für ihr Quadrat der Oppositionen.

- P und Q sind *kontradiktorisch*, wenn stets eine von ihnen falsch und eine von ihnen wahr ist.
- P und Q sind *konträr*, wenn niemals beide zugleich wahr sind (aber zugleich falsch sein dürfen).
- P und Q sind *subkonträr*, wenn niemals beide zugleich falsch sind.

Es gilt dann:

- /`Alle Kinder sind Rechtshänder.`/ und /`Kein Kind ist Rechtshänder.`/ sind kontradiktorisch. Ist das erste wahr, so ist das zweite falsch. Denn dann gibt es ein Kind, und dies ist Rechtshänder. Ist das erste falsch, so gibt es entweder keine Kinder oder alle Kinder die es gibt sind keine Rechtshänder. Also ist der zweite Satz wahr.
- /`Alle Kinder sind Rechtshänder.`/ und /`Einige Kinder sind keine Rechtshänder.`/ sind konträt. Sie können nicht zugleich wahr sein. Denn dann existieren Kinder und diese sind alle Rechtshänder, während zugleich mindestens eines von ihnen kein Rechtshänder ist. Widerspruch.
- /`Einige Kinder sind Rechtshänder.`/ und /`Einige Kinder sind keine Rechtshänder.`/ sind subkonträr. Sind beide falsch, so gibt es Kinder (sonst wäre der zweite Sat wahr). Und wenn es Kinder gibt, so sind sie alle keine Rechtshänder (weil der erste Satz falsch ist) und ebenso sind all Rechtshänder (weil der zweite falsch ist). Es gibt dann mindestens ein Kind, das sowohl Rechtshänder wie kein Rechtshänder ist. Widerspruch.

Übungen

[103] Überlegen Sie, ob die folgenden Schlüsse korrekt sind. Geben Sie ein Beispiel, wenn sie nicht gültig sind.

$$\begin{array}{ll} & \texttt{Kein } B \texttt{ ist } C\texttt{.} \\ & \texttt{Alle } A \texttt{ sind } B\texttt{.} \\ \hline \therefore & \texttt{Kein } A \texttt{ ist } C\texttt{.} \end{array} \tag{29.12a}$$

(29.12b)
$$\begin{array}{ll} & \texttt{Manche } B \texttt{ sind } C. \\ & \texttt{Manche } A \texttt{ sind nicht } B. \\ \hline \therefore & \texttt{Alle } A \texttt{ sind } C. \end{array}$$

(29.12c)
$$\begin{array}{ll} & \texttt{Kein } B \texttt{ ist } C. \\ & \texttt{Manche } A \texttt{ sind } B. \\ \hline \therefore & \texttt{Kein } A \texttt{ ist } C. \end{array}$$

[104] Der Modus Cesare geht wie folgt.

(29.13)
$$\begin{array}{ll} & \texttt{Kein } B \texttt{ ist } C. \\ & \texttt{Alle } A \texttt{ sind } C. \\ \hline \therefore & \texttt{Kein } A \texttt{ ist } B. \end{array}$$

Leiten Sie diesen mit Hilfe der Schlussregeln aus dem Modus Celarent ab.

Index

Äquivalenzrelation, 35
Äquivalenz
  semantische, 244
Übergangsrelation, 97

Ableitung, 79
  linksseitige, 123
Absorption, 155
abzählbar unendlich, 72
Alphabet, 61
  nichtterminales, 82
  terminales, 82
Assoziativität, 21
Assoziativität, 155
Asymmetrie, 35
Ausdruck, 210
  regulärer, 93
Ausdrucksträger, 135
Ausprägung, 136
Aussage, 149
  einfache, 144
  falsche, 142
  komplexe, 144
  wahre, 142
Automat, 97
  Sprache, 100
  totaler, 100
Axiom, 185, 186
Axiomatisierung, 233

Bedeutung, 135
Belegung, 218
Bild
  direktes, 40
Bindung, 243

C-ableitbar, 189, 249

de Morgan, 155
Deduktionstheorem, 181
Definitionsbereich, 39
Disjunktion, 140
  minimale, 166
Distributivgesetz, 22
Distributivität, 155
dreistellig, 207

Einfügung, 79
einstellig, 207
Element, 129
endlicher Automat, 97
Erfüllung, 151
Ersetzungsregel, 77
Euler-Venn Diagramm, 19
Existenzpräsupposition, 254
Exponentiation, 49

Folgerung, 177, 178
Formel, 208
  atomare, 206
Funktion, 39, 205
  bijektive, 42
  charakteristische, 45

injektive, 42
surjektive, 42
Funktionalsignatur, 207
Funktionswert, 39

Gegenstand
erster, 15
Gleichwertigkeit, 154
Grammatik, 82, 137
kontextfreie, 113
reguläre, 107

H-ableitbar, 187, 249
Hauptsymbol, 144
Hilbert Kalkül, 185
Hintereinanderschreibung, 63

Idempotenz, 21
Identität, 45
Implikation, 140
Induktion, 52
Induktionsanfang, 53
Induktionsschritt, 53
Irreflexivität, 35
Isomorphie, 234

Kardinalzahl, 57, 71
Klammerbilanz, 144
Klasse
axiomatisierbare, 233
Koinzidenzlemma, 154, 220, 226
Kommutativität, 21
Kommutativität, 155
Komplement
relatives, 19
Komposition, 44
Konjunktion, 140
minimale, 161
Konklusion, 177
Konsistenz, 192
Konstante, 205, 209
Konstituente, 116
Kontingenz, 151
Kontinuum, 73
konträr, 257
Kontradiktion, 151
kontradiktorisch, 257
Korrektheit, 188
Kripke-Rahmen, 197

Lauf, 99
akzeptierender, 99
Linearität, 35
Literal, 160

Mächtigkeit, 70
Menge, 11
leere, 13
transitive, 59
Metavariable, 241
Modus, 252
Modus Ponens, 179

Nachbereich, 41
Name, 205
Normalform, 126
Chomskysche, 126
disjunktive, 161
konjunktive, 166
Nullmenge, 13

Obersatz, 254
Objektbereich, 215
Operationszeichen, 140
Ordinalzahl, 57, 71

Paar, 31
Partition, 36

Postfix, 65
Präfix, 65
Prädikat, 203
Prämisse, 177
Produkt, 33, 89
    kartesisches, 33
Projektion, 32

Quantor, 206, 243

Reflexivität, 35
Regel, 178, 185
    kontextfreie, 113
    kontextgebundene, 80
Regelsystem, 81
Rekursion, 49
Relation, 33, 205
    konverse, 34
    symmetrische, 35
Relationalsignatur, 207
Relationenprodukt, 34
relatives Komplement, 25

Satz, 226
Schema, 136
Schluss, 178
    Gültigkeit, 178
Schnitt, 19
Semantik, 138
Signatur, 207
Skopus, 243
Sprache, 66
    kontextfreie, 113
    reguläre, 95
Startsymbol, 82
subkonträr, 257
Substitution, 181
Summenfolge, 54
Syllogistik, 251
Symmetrie, 35
Synonymität, 154
Syntax, 138

Tableau
    schließendes, 170
Tableau-Kalkül, 169
Tautologie, 151
Teilmenge, 23
    echte, 23
Teilwort, 65
Term, 207
    konstant, 221
Tilgung, 79
Transitivität, 35
Typ, 136

Universum, 131
Untersatz, 254
Urteilszeichen, 157

Variable, 16, 149
Vereinigung, 19
Verkettung, 63, 89
Vollständigkeit, 188
Vorbereich, 41
Vorkommen, 66, 136
    freies, 243
    gebundenes, 243
    uneigentlich, 243

Wahrheitstafel, 162
Welt, 197
Wert, 39
Wertebereich, 39
Wort, 61
    Länge, 62
    leeres, 63
Wurzel, 117

Zählreihe, 57
Zeichen, 135
Zeichenkette, 61
    Länge, 62
Zeichensubstanz, 135
Zeichensystem, 135
Zugänglichkeitsrelation, 197
Zustand, 97
    finaler, 97
    initialer, 97
zweistellig, 207

Literatur

Bauer, Friedrich und Martin Wirsing. *Elementare Aussagenlogik*. Springer, 1991.

Boolos, George, John P. Burgess und Richard C. Jeffrey. *Computability and Logic*. Cambridge University Press, 2002.

Carstensen, K.-U. u. a. *Computerlinguistik und Sprachtechnologie. Eine Einführung*. 2. Aufl. Berlin: Spektrum. Akademischer Verlag, 2004.

Kleinknecht, Reinhard und Ekkehard Wüst. *Lehrbuch der elementaren Logik*. Wissenschaftliche Reihe. dtv, 1976.

Langendoen, D. Terence und Paul M. Postal. *The Vastness of Natural Language*. Oxford: Basil Blackwell, 1984.

Lyndon, Robert C. *Notes on Logic*. Van Nostrand, 1967.

Parsons, Terence. *Articulating Medieval Logic*. Oxford University Press, 2014.

Partee, Barbara, Alice ter Meulen und Robert Wall. *Mathematical Methods in Linguistics*. Dordrecht: Kluwer, 1990.

Rautenberg, Wolfgang. *Einführung in die Mathematische Logik. Ein Lehrbuch mit Berücksichtigung der Logikprogrammierung*. Braunschweig/Wiesbaden: Vieweg, 1996.

— *Klassische und nichtklassische Aussagenlogik*. Braunschweig/Wiesbaden: Vieweg Verlag, 1979.

Schöning, Uwe. *Logik für Informatiker*. 5. Aufl. Spektrum Verlag, 2000.