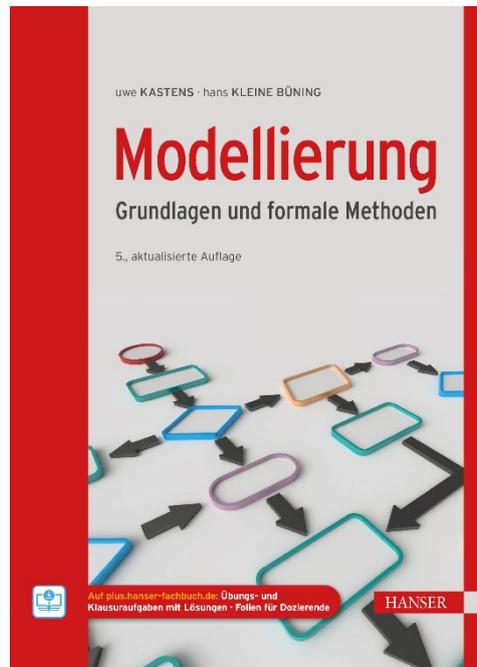


HANSER



Leseprobe

zu

Modellierung

von Uwe Kastens und Hans Kleine Büning

Print-ISBN: 978-3-446-46942-6

E-Book-ISBN: 978-3-446-46956-3

Weitere Informationen und Bestellungen unter

<https://www.hanser-kundencenter.de/fachbuch/artikel/9783446469426>

sowie im Buchhandel

© Carl Hanser Verlag, München

Vorwort

Das Modellieren ist eine für das Fach Informatik typische Arbeitsmethode, die in allen Gebieten des Faches angewandt wird. Aufgaben, Probleme oder Strukturen werden untersucht und als Ganzes oder in Teilaspekten beschrieben, bevor sie durch den Entwurf von Software, Algorithmen, Daten oder Hardware gelöst bzw. implementiert werden. Mit der Modellierung einer Aufgabe zeigt man, ob und wie sie verstanden wurde. Das Modell ist Voraussetzung und Maßstab für die Lösungen und liefert meist auch den Schlüssel für einen systematischen Entwurf. Als Ausdrucksmittel für die Modellierung steht ein breites Spektrum von Kalkülen und Notationen zur Verfügung. Sie sind spezifisch für unterschiedliche Arten von Aufgaben und Problemen. Deshalb werden in den verschiedenen Gebieten der Informatik unterschiedliche Modellierungsmethoden eingesetzt. In den entwurfsorientierten Gebieten, wie Software-Technik und Hardware-Entwurf, ist die Bedeutung der Modellierung und die Vielfalt der Methoden besonders stark ausgeprägt.

Mit diesem Buch soll eine Übersicht über die wichtigsten Kalküle der Informatik und ein grundlegendes Verständnis für jeden der vorgestellten Kalküle vermittelt werden. Die Leser sollen an vielen praktischen Beispielen lernen, die Kalküle zur Modellierung anzuwenden, und dabei Erfahrungen im formalen Beschreiben erwerben. Sie sollen den Nutzen von klaren und präzisen Beschreibungen erkennen. Das angebotene Spektrum von Kalkülen ist als Grundausrüstung zu verstehen, die noch wesentlich vertieft und verbreitert werden kann. Deshalb wird hier von jedem Kalkül nur der innere methodische Kern präsentiert und auf die Vorstellung von Erweiterungen und weniger grundlegenden Kalkülen verzichtet.

Das vorliegende Buch ist als Lehrbuch zur Vorlesung *Modellierung* entstanden, mit der die oben genannten Ziele verfolgt werden. Die Vorlesung wird an der Universität Paderborn als eine einsemestrige, vierstündige Pflichtveranstaltung für Studierende der Informatik und der Wirtschaftsinformatik angeboten. Zur Vorlesung gehören praktische Übungen in Form betreuter Kleingruppenarbeit, selbstständiger Hausarbeiten und zentraler Präsentation von Lösungen. Die Übungsaufgaben dieses Buches stammen im Wesentlichen aus diesem Übungsmaterial.

Die beiden Autoren dieses Buches haben die Vorlesung in Abstimmung mit ihren Kollegen inhaltlich konzipiert und seit 1998 im zweijährigen Wechsel gehalten. Wegen ihres unterschiedlichen fachlichen Hintergrundes setzen sie auch verschiedene Schwerpunkte bei der Vermittlung von Modellierungsaspekten: Uwe Kastens arbeitet im Gebiet Programmiersprachen und Übersetzer und ist stark in Themen der Software-Technik verwurzelt. Bei der Modellierung betont er die Anwendung der Kalküle, den Nutzen formaler Beschreibungen und die Trennung von Aufgaben und Lösungen. Hans Kleine Büning ar-

beitet im Gebiet Logik und Wissensbasierte Systeme. Er betont stärker die theoretischen Grundlagen der Kalküle und setzt Schwerpunkte in den Logik-Kalkülen. Die Autoren haben in enger Kooperation versucht, die unterschiedlichen Schwerpunkte und Herangehensweisen einander ergänzend in das Buch und die Vorlesung einzubringen. An diesen Diskussionen hat sich auch der Kollege Hauenschild nachhaltig beteiligt, der im Jahr 2004 die Vorlesung übernommen hat.

Nach der Einführung werden in den Kapiteln 2 bis 7 die Kalküle vorgestellt. Wir haben die Reihenfolge so gewählt, dass Vorgriffe möglichst vermieden werden können. Zu jedem Kalkül werden zunächst die Grundbegriffe eingeführt und dann typische Modellierungstechniken an möglichst anschaulichen Beispielen gezeigt. Jedes Kapitel schließt mit einer Sammlung von Übungsaufgaben. Als durchgängige Aufgabe wird die Modellierung von Aspekten eines Getränkeautomaten in Übungen eines jeden Kapitels aufgegriffen. In Kapitel 8 werden an zwei Fallstudien alle Kalküle im Zusammenhang gezeigt. Einige bibliografische Hinweise finden sich am Ende des Buches.

Im Kapitel 1 wird der Modellbegriff eingeführt, so wie er im Alltag und in der Informatik verwendet wird. Die Tätigkeit des Modellierens wird charakterisiert und ihre Notwendigkeit begründet. Am Schluss zeigen wir eine einfache formale Modellierung im Vorgriff auf später vorgestellte Kalküle.

Das Kapitel 2 führt Mengen als Modellierungskalkül ein: Einfache und zusammengesetzte Mengen definieren die Wertebereiche von Objekten und Eigenschaften des Modells. Abstrakte Konzepte wie kartesisches Produkt, Potenzmenge, Vereinigung, Folgen, Relationen und Funktionen werden zur Strukturierung der Wertemengen eingesetzt. Diese Begriffe sind Grundlagen für jede Art formaler Beschreibung.

Fast alle formalen Kalküle definieren eine Notation für Formeln, in denen Operanden mit Operationen verknüpft werden. Im Kapitel 3 werden deshalb Terme als abstrakte Grundlage von Formeln eingesetzt, die erst im Anwendungskontext spezielle Bedeutung bekommen. Notationen sowie allgemein gültige Begriffe wie Substitution, Umformung nach Regeln und Unifikation werden hier universell für Terme definiert. Im zweiten Teil des Kapitels werden diese Begriffe zu abstrakten und konkreten Algebren ausgebaut. Damit können dann Kalküle und Datenstrukturen algebraisch definiert werden. Wegen der universellen Bedeutung von Termen haben wir sie in einem separaten Kapitel definiert – und nicht als Teil der Prädikatenlogik, wo sie einen angestammten Platz haben.

Kapitel 4 führt die beiden klassischen Gebiete der Logik ein: Aussagenlogik und Prädikatenlogik erster Stufe. Wir beschränken uns auf die für Anfänger wichtigsten Grundlagen der Kalküle: Syntax und Semantik, Umgang mit logischen Operatoren und Quantoren und einfache Normalformen sowie Transformationen. Insbesondere Kalküle und Verfahren des automatischen Beweisens überlassen wir einer späteren Vertiefung.

Der Kalkül der Graphen wird in Kapitel 5 behandelt. Er eignet sich besonders gut zum Modellieren. Er ist leicht zu verstehen, da er nur auf Relationen basiert, und er ist anschaulich, da er sehr suggestive Visualisierungen hat. Graphen sind außerordentlich vielfältig einsetzbar, da Objekte und Beziehungen zwischen Objekten in so vielen unterschiedlichen Bedeutungen beim Modellieren vorkommen. Es werden hier nur die grund-

legenden Eigenschaften von Graphen vorgestellt. Auf weiterführende Begriffe der Graphentheorie wird verzichtet und stattdessen die Vielfalt der Einsatzgebiete und Modellierungstechniken ausgebreitet.

In Kapitel 6 führen wir zwei grundlegende Kalküle ein, die sich besonders zur Modellierung struktureller Eigenschaften eignen. Kontextfreie Grammatiken werden als Regelsystem vorgestellt, mit dem Baumstrukturen definiert werden können, um damit geschachtelte Strukturen zu modellieren. Die Definition von Sprachen als Menge von Symbolfolgen, die sonst im Vordergrund steht, spielt hier nur eine Nebenrolle. Im zweiten Teil wird das Entity-Relation-Ship-Modell eingeführt. Damit formuliert man Regeln, nach denen Systeme in Mengen gleichartiger Objekte mit bestimmten Eigenschaften gegliedert sind und zwischen denen Relationen bestehen. Das ER-Modell ist Grundlage sowohl der Schemata objektorientierter Datenbanken als auch der Spezifikation von Strukturen und Beziehungen in Software-Systemen, z. B. mit der Modellierungssprache UML.

In Kapitel 7 führen wir zwei grundlegende Kalküle ein, mit denen Abläufe modelliert werden können: endliche Automaten und Petri-Netze. Sie werden eingesetzt, um das dynamische Verhalten von Systemen zu beschreiben. Man gibt die Zustände an, die das System einnehmen kann, und beschreibt, unter welchen Bedingungen es aus einem Zustand in einen anderen übergehen kann. Beide Kalküle sind mit recht einfachen Regeln definiert und haben sehr anschauliche grafische Repräsentationen. Mit endlichen Automaten werden meist sequentielle, mit Petri-Netzen nebenläufige Prozesse modelliert. Beide dienen als Grundlage komplexer Kalküle (z. B. endliche Automaten für Statecharts), die hier nicht vorgestellt werden.

In Kapitel 8 präsentieren wir die Modellierungen von zwei Aufgaben vollständig im Zusammenhang: Die Auftragsabwicklung in einer Autowerkstatt und Regeln eines Gesellschaftsspiels. Verschiedene Aspekte des Gegenstandsbereiches werden mit jeweils passenden Kalkülen modelliert. Außerdem werden einzelne Aspekte zum Vergleich mit verschiedenen Kalkülen beschrieben. Dabei wird das Vorgehen ausführlich erläutert. Wir zeigen damit, dass es wichtig ist, den Gegenstandsbereich sinnvoll zu zerlegen und für Teilaspekte geeignete Kalküle zu wählen.

Zu diesem Buchprojekt haben die Autoren nachhaltige Unterstützung erfahren: Kollege Hauenschild hat im Jahr 2004 unser Material zur Vorbereitung der Vorlesung übernommen. Er hat uns viele nützliche Hinweise gegeben und zur Behebung einiger Inkonsistenzen beigetragen. Wissenschaftliche Mitarbeiter haben die Übungen zur Vorlesung betreut. Stellvertretend seien hier Dinh Khoi Le, Jochen Kreimer, Theodor Lettmann und Carsten Schmidt genannt. Sie haben sich viele Aufgaben fantasievoll ausgedacht, das Vorlesungsmaterial kritisch hinterfragt und Fehler darin korrigiert. Sigrid Gundelach hat den größten Teil des Manuskriptes geschrieben, spröde Formeln gewissenhaft übertragen und sich engagiert in das manchmal widerspenstige Textsystem eingearbeitet. Michael Thies hat uns geholfen, mit dem System den gewünschten Drucksatz zu produzieren. Sie alle haben zum Gelingen des Buches beigetragen. Wir danken ihnen sehr dafür.

Uwe Kastens, Hans Kleine Büning

Inhalt

| | | |
|----------|--|-----------|
| 1 | Einführung | 15 |
| 1.1 | Einführendes Beispiel | 15 |
| 1.2 | Modellbegriff | 18 |
| | Übungen | 23 |
| 2 | Modellierung mit Wertebereichen | 25 |
| 2.1 | Mengen | 27 |
| 2.2 | Potenzmengen | 29 |
| 2.3 | Kartesische Produkte | 30 |
| 2.4 | Vereinigung | 32 |
| 2.5 | Folgen | 33 |
| 2.6 | Relationen | 34 |
| 2.7 | Funktionen | 38 |
| 2.8 | Beispiel im Zusammenhang | 42 |
| 2.9 | Fallstudie: Getränkeautomat | 44 |
| | 2.9.1 Produkte und Vorrat | 45 |
| | 2.9.2 Kassieren | 46 |
| | 2.9.3 Bedienung und Zustand | 46 |
| | Zusammenfassung | 49 |
| | Übungen | 49 |
| 3 | Terme und Algebren..... | 57 |
| 3.1 | Terme | 58 |
| | 3.1.1 Sorten und Signaturen | 58 |
| | 3.1.2 Notationen für Terme | 61 |
| 3.2 | Substitution und Unifikation | 64 |
| | 3.2.1 Substitution | 65 |
| | 3.2.2 Unifikation | 68 |
| 3.3 | Algebren | 70 |
| | 3.3.1 Abstrakte Algebra | 71 |
| | 3.3.2 Konkrete Algebra | 72 |
| 3.4 | Algebraische Spezifikation von Datenstrukturen | 73 |
| 3.5 | Algebraische Spezifikation für den Getränkeautomaten | 80 |
| | Übungen | 81 |

| | | |
|----------|---|------------|
| 4 | Logik | 87 |
| 4.1 | Aussagenlogik | 88 |
| 4.1.1 | Syntax der Aussagenlogik | 88 |
| 4.1.2 | Semantik der Aussagenlogik | 89 |
| 4.1.3 | Normalformen | 95 |
| 4.1.4 | Aussagenlogische Modellbildung | 99 |
| 4.2 | Prädikatenlogik | 100 |
| 4.2.1 | Syntax der Prädikatenlogik | 100 |
| 4.2.2 | Semantik der Prädikatenlogik | 104 |
| 4.2.3 | Normalformen | 109 |
| 4.2.4 | Modellbildung mit der Prädikatenlogik | 113 |
| 4.3 | Elementare Beweistechniken | 116 |
| 4.3.1 | Formaler Rahmen und Grundlagen | 117 |
| 4.3.2 | Elementare Beweisstrukturen | 120 |
| 4.3.3 | Quantoren | 123 |
| | Übungen | 130 |
| 5 | Modellierung mit Graphen | 135 |
| 5.1 | Grundlegende Definitionen | 136 |
| 5.2 | Wegeprobleme | 143 |
| 5.3 | Verbindungsprobleme | 152 |
| 5.4 | Modellierung mit Bäumen | 156 |
| 5.5 | Zuordnungsprobleme | 164 |
| 5.6 | Abhängigkeiten | 168 |
| | Übungen | 176 |
| 6 | Modellierung von Strukturen | 181 |
| 6.1 | Kontextfreie Grammatiken | 183 |
| 6.2 | Baumstrukturen in XML | 194 |
| 6.2.1 | XML Notation | 195 |
| 6.2.2 | XML-Texte als Bäume | 197 |
| 6.2.3 | Strukturdefinition für XML-Bäume | 200 |
| 6.3 | Entity-Relationship-Modell | 202 |
| 6.3.1 | Entity-Mengen | 203 |
| 6.3.2 | Attribute | 204 |
| 6.3.3 | Relationen | 206 |
| 6.4 | Klassendiagramme in UML | 215 |
| 6.4.1 | Klassen mit Attributen | 216 |
| 6.4.2 | Assoziationen | 216 |
| | Übungen | 221 |
| 7 | Modellierung von Abläufen | 227 |
| 7.1 | Endliche Automaten | 228 |
| 7.1.1 | Zeichenfolgen über Alphabete | 230 |

| | | |
|----------|--|------------|
| 7.1.2 | Deterministische endliche Automaten | 232 |
| 7.1.3 | Nicht-deterministische endliche Automaten | 234 |
| 7.1.4 | Endliche Automaten mit Ausgabe | 238 |
| 7.1.5 | Endliche Automaten in UML | 241 |
| 7.2 | Petri-Netze | 244 |
| | Übungen | 256 |
| 8 | Fallstudien..... | 263 |
| 8.1 | Fallstudie Autowerkstatt | 263 |
| 8.1.1 | Informationsstruktur und Zusammenhänge | 264 |
| 8.1.2 | Bedingungen und Regeln | 267 |
| 8.1.3 | Abläufe der Auftragsbearbeitung | 268 |
| 8.2 | Fallstudie Gesellschaftsspiel | 270 |
| 8.2.1 | Strukturen und Zusammenhänge | 270 |
| 8.2.2 | Bedingungen und Regeln | 274 |
| 8.2.3 | Spielabläufe | 275 |
| 8.3 | Fallstudie: Aussagenlogische Formeln transformiert durch Graphen | 277 |
| 8.3.1 | Erfüllbarkeitsäquivalente Transformation in KNF 277 | |
| 8.3.2 | Beispielanwendung | 282 |
| | Übungen | 283 |
| | Bibliographie..... | 287 |
| | Referenzen | 290 |
| | Register | 293 |
| | Auflistung Zusatzmaterial | 301 |

1

Einführung

Das Anfertigen von Modellen ist ein wichtiger Schritt in den frühen Phasen der Herstellung ganz unterschiedlicher Objekte: Ein Architekt erstellt ein dreidimensionales Modell für den Neubau eines besonders exponierten Gebäudes. Daran werden mit dem Auftraggeber die äußere Gestaltung und die Integration des Gebäudes in das Stadtbild diskutiert. Um die Raumaufteilung zu planen, die Sicherheit zu prüfen und die Durchführung des Baus zu steuern, werden andere Modelle erstellt, wie Bauzeichnungen und Berechnungen der Statik.

Solch ein Vorgehen ist auch typisch für Entwicklungsprozesse in der Informatik: Aufgaben und Lösungsvorschläge werden mit Modellen beschrieben, bevor mit der Herstellung einer Lösung in Software oder Hardware begonnen wird. An den Modellen können Auftraggeber und Entwickler prüfen, ob die Aufgabenstellung richtig verstanden wurde, welche Eigenschaften die vorgeschlagene Lösung haben wird und ob diese akzeptabel sind.

Jede Branche hat Techniken der Modellbildung, die sich für ihre Aufgaben besonders gut eignen, z. B. Gebäudemodelle, Bauzeichnungen und Berechnungen der Statik für die Herstellung von Gebäuden oder Schaltpläne für das Anfertigen elektrischer Anlagen. Produkte der Informatik wirken meist durch ihre Funktionen – seltener durch ihr Aussehen. Deshalb überwiegen hier abstrakte Modelle, die mit formalen Kalkülen erstellt werden. Aufgaben von unterschiedlicher Art aus einem sehr breiten Spektrum werden mit Informatikmethoden gelöst. Deshalb werden in der Informatik sehr viele verschiedene Kalküle eingesetzt, die jeweils unterschiedliche Aspekte der Aufgaben möglichst gut modellieren können. Informatiker müssen sie als ihr „Handwerkszeug“ beherrschen, um für ihre Aufgaben nützliche Modelle mit geeigneten Kalkülen anzufertigen.

1.1 Einführendes Beispiel

Wir wollen einen ersten Einblick in das Modellieren der Informatik geben. Als Beispiel betrachten wir folgende Denksportaufgabe:

Beispiel 1.1: Flussüberquerung

Ein Mann steht mit einem Wolf, einer Ziege und einem Kohlkopf am linken Ufer eines Flusses, den er überqueren will. Er hat ein Boot, das gerade groß genug ist, ihn und ein weiteres Objekt zu transportieren, sodass er immer nur eines der drei mit sich hinübernehmen kann. Falls der Mann allerdings den Wolf mit

der Ziege oder die Ziege mit dem Kohlkopf unbewacht an einem Ufer zurücklässt, wird einer gefressen werden.

Ist es möglich, den Fluss zu überqueren, ohne dass die Ziege oder der Kohlkopf gefressen wird?

Solche Aufgaben sind deswegen reizvoll, weil man eine Weile daran herumknobelt und dann – wie mit einem Geistesblitz – die Lösung findet. Mit solch einem Vorgehen geht man allerdings das Risiko ein, die Lösung nicht zu finden. Bei ernsthaften Aufgaben wäre das fatal. Deshalb zeigen wir hier ein systematisches Vorgehen unter Einsatz von Informatik-Kalkülen – auch wenn dabei vielleicht der Spaß an der Knobelei verloren geht.

Zunächst klären wir, welche Aufgabe gelöst werden soll: Die Antwort auf die Frage am Ende des Textes kann „ja“ oder „nein“ lauten. In jedem Fall müssen wir eine Begründung angeben, entweder indem wir einen Plan für eine sichere Flussüberquerung entwickeln oder erklären, weshalb es einen solchen nicht gibt.

Dann beginnen wir die Modellierung, indem wir die Beschreibung der Aufgabe auf die unbedingt notwendigen Objekte, Eigenschaften und Aktionen reduzieren: Als Objekte kommen in Frage: Mann, Wolf, Ziege, Kohlkopf, Fluss, linkes Ufer und rechtes Ufer, Boot. Es ist wichtig, als Eigenschaft zu beschreiben, welche der Objekte Wolf, Ziege und Kohlkopf sich gemeinsam an jedem der beiden Ufer aufhalten; denn damit wird entschieden, ob einer von ihnen gefressen werden kann. Als Aktionen benötigen wir das Transportieren jeweils eines der drei Objekte von einem zum anderen Ufer oder Leerfahrten jeweils in beide Richtungen.

Wenn es eine Lösung der Aufgabe gibt, dann kann man sie als eine Folge solcher Fahrten über den Fluss beschreiben. Jede Fahrt verändert den Zustand des Systems. Die Zustände des Systems modellieren wir, indem wir angeben, welche der Objekte Mann, Wolf, Ziege und Kohlkopf sich an welchem Ufer befinden. Formal ist der Zustand ein Paar von Mengen (l, r) , mit $l, r \subseteq \{M, W, Z, K\}$, wobei wir Abkürzungen für die vier Objekte verwenden. Da keines der Objekte verschwinden und keines hinzukommen kann, muss für jedes Paar (l, r) gelten

$$l \cap r = \emptyset \quad \text{und} \quad l \cup r = \{M, W, Z, K\}$$

Nun können wir auch formal beschreiben, wann ein Zustand (l, r) unzulässig ist, weil die Ziege oder der Kohlkopf gefressen werden könnte:

$$\begin{aligned} M \notin l \text{ und } (\{W, Z\} \subseteq l \text{ oder } \{Z, K\} \subseteq l) \text{ oder} \\ M \notin r \text{ und } (\{W, Z\} \subseteq r \text{ oder } \{Z, K\} \subseteq r) \end{aligned}$$

Wenn z. B. im Zustand $(\{M, W, Z, K\}, \emptyset)$ die Ziege an das rechte Ufer transportiert wird, geht das System in den Zustand $(\{W, K\}, \{M, Z\})$ über. Beide Zustände sind gemäß obiger Bedingung zulässig. Wir müssen noch begründen, dass es nicht nötig ist, den Zustand während der Überfahrt zu modellieren: Ist während der Überfahrt das gerade verlassene Ufer unzulässig, so ist es auch unzulässig im Zustand, der die Ankunft modelliert. Entsprechendes gilt für den Zustand vor der Abfahrt und das zu erreichende Ufer.

Die Zustandsübergänge kann man sehr anschaulich grafisch beschreiben. Abb. 1.1 zeigt den Übergang, den wir oben als Beispiel beschrieben haben. Die Rechtecke geben die Zustände mit dem Paar von Mengen an. Der beschriftete Pfeil verbindet die Zustände vor und nach dem Transport.

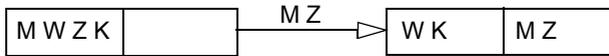


Abbildung 1.1: Ein Zustandsübergang

Wir haben nun das Prinzip des Modells entworfen. Wir brauchen es nur noch vollständig auszufüllen, indem wir alle zulässigen Zustände und Übergänge angeben. Abb. 1.2 zeigt das vollständige Modell. Es sind auch einige unzulässige Zustände angegeben, um zu zeigen, welche Entscheidungen nicht getroffen werden dürfen. Die Übergänge zwischen zulässigen Zuständen sind natürlich jeweils in beiden Richtungen möglich, obwohl es im Sinne der Aufgabe nicht sinnvoll ist, in einen früheren Zustand zurück zu rudern. In Abb. 1.2 können wir nun leicht ablesen, dass es mehrere Lösungen für die Transportaufgabe gibt:

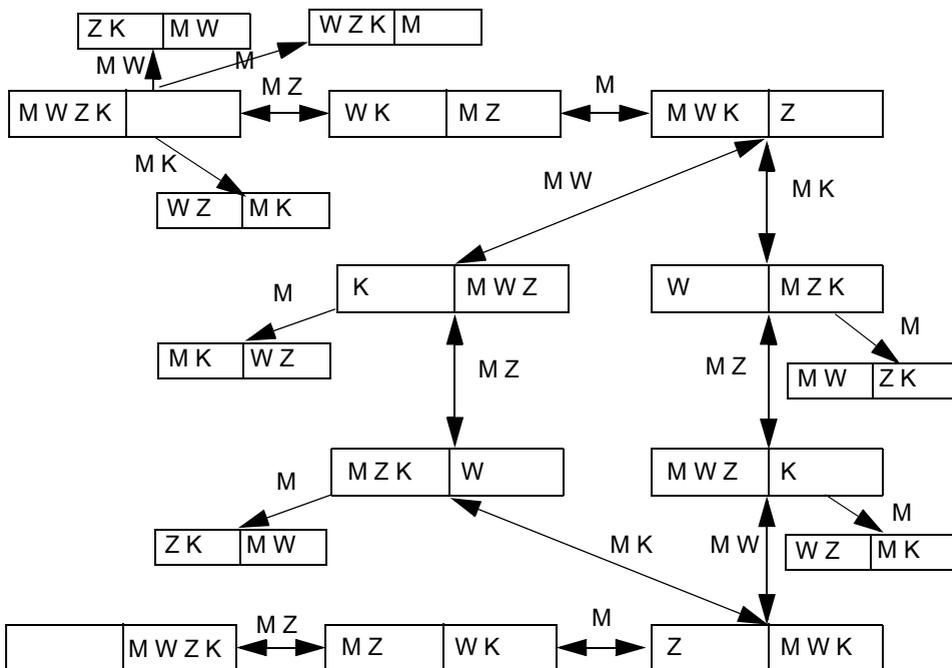


Abbildung 1.2: Modellierung der Flussüberquerung

Jeder Weg vom Startzustand zum Zielzustand beschreibt eine Lösung. Es gibt zwei verschiedene, die mit 7 Überfahrten auskommen. Alle anderen Transportfolgen enthalten überflüssige Überfahrten.

Für die Modellierung unserer Aufgabe haben wir den Kalkül der endlichen Automaten angewandt. Er eignet sich besonders gut, wenn Abläufe in Systemen mit Übergängen zwischen verschiedenen Zuständen beschrieben werden sollen. Der Kalkül wird in Abschnitt 7.1.2 vorgestellt. Die grafische Veranschaulichung des Automaten in Abb. 1.2 benutzt den Kalkül der gerichteten Graphen, der in Abschnitt 5.1 eingeführt wird. Zu Beginn der Modellierung haben wir die Zustände des Systems durch Paare von Mengen beschrieben. Solche Abstraktionen führen wir in Kapitel 2 ein.

In den folgenden Kapiteln dieses Buches werden wir die Kalküle jeweils an typischen Beispielen vorstellen. Eine Modellierungsaufgabe werden wir dabei durchgängig immer wieder aufgreifen: Die Bedienung eines Getränkeautomaten soll modelliert werden. Das Gerät soll Getränke wie Kaffee, Tee und Kakao gegen Bezahlung mit Münzen abgeben. Man soll Varianten der Getränke wählen können, z. B. mit oder ohne Milch oder Zucker. Die Modellierung soll berücksichtigen, dass im Gerät nur begrenzte Vorräte für die Zubereitung der Getränke untergebracht werden können. Die Beschreibung der Aufgabe ist hier absichtlich unscharf gehalten. Es entspricht dem Vorgehen in der Realität, dass die Beschreibung der Aufgabe erst im Zuge der Modellierung präzisiert wird.

1.2 Modellbegriff

Der Begriff des *Modells* ist vom lateinischen Wort *modulus* für Maß, Maßstab abgeleitet. Er wird in vielen verschiedenen Zusammenhängen mit recht unterschiedlichen Bedeutungen verwendet. Er kann das Abbild eines vorhandenen Originals bezeichnen, z. B. ein Schiffsmodell, oder das Vorbild für ein herzustellendes Original, z. B. ein Gebäudemodell oder ein Vorbild für Maler oder Bildhauer. Das Modell kann konkret sein, wie das Schiffsmodell, oder abstrakt, wie ein Modell zur Rentenberechnung. Auch das Modellerte kann konkret sein, wie das Schiff, oder abstrakt, wie die zahlenmäßige Entwicklung der Bevölkerung. In Abb. 1.3 haben wir die Erklärung des Modellbegriffes aus einem allgemeinen Lexikon angegeben. Für unsere Zwecke ist darin die Variante zum „Sprachgebrauch verschiedener Wissenschaften“ zutreffend. Die Modelle in der Informatik sind im Allgemeinen abstrakte Abbilder oder Vorbilder zu konkreten oder abstrakten Originalen. So ist das Modell der Flussüberquerung im vorigen Abschnitt ein abstrakter, endlicher Automat. Auch das Modellerte ist in diesem Beispiel abstrakt: der schrittweise Ablauf der Flussüberquerung mit den dabei durchlaufenen Zuständen.

Man beachte, dass in der Logik, einem Teilgebiet der Mathematik und Informatik, der Begriff Modell mit einer sehr speziellen, anderen Bedeutung verwendet wird: *Eine Struktur S ist ein Modell der logischen Formeln F , wenn alle Formeln aus F für S gelten* (siehe auch Kapitel 4).

Modell [italien., zu lat. *modulus* „Maß, Maßstab“], allg. Muster, Vorbild, Entwurf.
 – Mensch (auch Tier), der (das) als Vorbild für künstler. Studien oder Kunstwerke dient („sitzt“).
 – in der Bildhauerei meist in verkleinerter Form ausgeführter Entwurf einer Plastik oder Tonarbeit, die in Bronze gegossen werden soll. - †Architekturmodell.
 – in der Modebranche Bez. für 1. ein nur einmal oder in eng begrenzter Anzahl hergestelltes Kleidungsstück. (*M.kleid*); 2. die Vorlage für eine Vervielfältigung; 3. svw. Mannequin.
 – im Sprachgebrauch verschiedener Wiss. (Philosophie, Naturwiss., Soziologie, Psychologie, Wirtschaftswiss., Politikwiss., Kybernetik u.a.) ein Objekt materieller oder ideeller (Gedanken-M.) Natur, das von einem Subjekt auf der Grundlage einer Struktur-, Funktions- oder Verhaltensanalogie für ein anderes Objekt (*Original*) eingesetzt und genutzt wird, um Aufgaben zu lösen, deren Durchführung unmittelbar am Original selbst nicht möglich bzw. zu aufwendig ist (z. B. Flugzeug-M. im Windkanal). Die **Modellmethode** vollzieht sich in vier Schritten: 1. Auswahl (Herstellung eines dem [geplanten] Original entsprechenden M.; 2. Bearbeitung des M., um neue Informationen über das M. zu gewinnen (**Modellversuch**; †Ähnlichkeitsgesetze); 3. Schluss auf Informationen über das Original (meist Analogieschluß); ggf. 4. Durchführung der Aufgabe am Original. Infolge der Relationen zw. Subjekt, Original und M. (**Modellsystem**) ist ein M. einsetzbar u. a. zur Gewinnung neuer Informationen über das Original (z. B. Atom-M.), zur Demonstration und Erklärung (z. B. Planetarium), zur Optimierung des Originals (z. B. Netzplan), zur Überprüfung einer Hypothese oder einer techn. Konstruktion (z. B. Laborversuch). - Abweichend von diesem M.begriff versteht die *mathemat. Logik* unter M. eine Interpretation eines Axiomensystems, bei der alle Axiome dieses Systems wahre Aussagen darstellen. Diese **Modelltheorie** liefert grundlegende Verfahren zur Behandlung von Fragen der Vollständigkeit, Widerspruchsfreiheit und Definierbarkeit.

Abbildung 1.3: Modellbegriff aus *Meyers Neues Lexikon* [20]

Modelle sind absichtlich nicht originalgetreu; sie heben bestimmte Eigenschaften hervor und lassen andere weg. Der intendierte Verwendungszweck des Modells bestimmt, welche Eigenschaften modelliert werden und welches Kalkül zu deren Beschreibung besonders geeignet ist. So werden beim Hausbau für verschiedene Zwecke ganz unterschiedliche Arten von Modellen verwendet:

- ein Gebäudemodell zur Vermittlung des optischen Eindruckes,
- ein Grundriss zur Einteilung der Räume und des Grundstückes,
- ein Kostenplan zur Finanzierung und
- ein Gewerkeplan zur Durchführung des Baus.

Register

Ziffern

- 0-stellig 40, 59
- 1-stellig 35, 40
- 2-stellig 36, 38

A

- Abhängigkeitsgraph 168, 268
- Ablaufgraph 173
- Ableitung 184, 185
- Ableitungsbaum 185, 200, 289
- Ableitungsschritt 184
- Abschnitt, kritischer 247
- abstrakte Syntax 164
- Adjazenzlisten 140
- Adjazenzmatrix 139
- Aggregation 219
- akzeptierte Sprache 233, 236
- Algebra 80, 287
 - abstrakte 71
 - Boolesche 71
 - konkrete 73
- allgemeingültig 92
- allgemeinster Unifikator 69
- Alphabet 230
 - Ausgabe- 238
 - Eingabe- 232
- Anfangselement 126
- Anfangsmarkierung 247
- Anfangs-Tag 195
- Anfangszustand 232, 242
- Anordnung 169
- antisymmetrisch 36, 37, 114
- Äquivalenz 88
 - logische 94
 - relation 37, 114
- arc 137
- Assoziation 216

- binäre 218
 - Assoziativität 95
 - asymmetrisch 36
 - Atom 89
 - Attribut 196, 202, 204, 205, 216, 217
 - Schlüssel- 206
 - Aufrufgraph 175
 - Ausdruck, regulärer 201, 230
 - Ausführungsreihenfolge 169
 - Ausgabealphabet 238
 - Ausgabefunktion 238
 - Ausgangsgrad 141
 - Aussagenlogik 88, 116, 288
 - Semantik der 89
 - Syntax der 88
 - Ausschluss, gegenseitiger 247, 256
 - Auszeichnungen 194
 - Auszeichnungssprachen 194
 - Automat
 - deterministischer 232, 268, 275
 - endlicher 18, 176, 227, 228, 232, 235, 241, 275, 289
 - Mealy- 238
 - Moore- 238
 - nicht-deterministischer 235
 - Teil- 241
 - vollständiger 233
 - Axiom 71
 - mit Axiomen umformen 72
 - azyklischer Graph 146
- ## B
- Baum 63
 - Binär- 158
 - darstellung 63
 - Entscheidungs- 159
 - gerichteter 157

- gewurzelter 185, 199
- Struktur- 164
- Teil- 157, 199
- ungerichteter 152
- Behauptung 117
- Beweis 116, 117
 - indirekter 117, 123
 - induktiver 126
 - struktur 120
- Bewertung 89
 - von Formeln 90
- bijektiv 40
- Bildbereich 38
- binär 218, 250
- Binärbaum 158
- Bindungsstärke 62
- bipartit 165
- Brückenkante 155
- C**
- charakteristische Funktion 40
- chromatische Zahl 168
- complete 220
- D**
- DAG 146
- De Morgan 95
- Definition 119
 - induktive 126
- Definitionsbereich 38
- deklarativ 22
- deterministischer endlicher Automat 232, 268, 275
- Diagrammtypen 215
- Differenz 28
- directed acyclic graph 146
- disjoint 220
- disjunkt 28
 - Teilmengen 59
- disjunkte Vereinigung 33
- Disjunktion 88, 120
- disjunktive Normalform 98
- Distributivität 95
- DNF 98
- Document Object Model 201
- Document Type Definiton 200
- DOM 201
- DTD 200
- duale Modellierung 172
- Durchschnitt 28
- E**
- edge 137
- Eingabealphabet 232
- Eingangsgrad 141
- Element 27
- endlicher Automat 18, 176, 227, 228, 232, 241, 275, 289
- End-Tag 195
- Endzustand 232, 242
- Entity 202
 - Menge 203, 216
 - Relationship-Modell 181, 202, 215, 264, 271, 289
- Entscheidungsbaum 159
- ER-Diagramm 203
- erfüllbar 91
- erfüllbarkeitsäquivalent 109
- ER-Modell 181, 202, 215, 264, 271, 289
- Euler 144
 - Kreis 148
 - Weg 148
- Exemplar 214
- Extensible Markup Language 194, 289
- F**
- Fallunterscheidung 118, 124
- falsifizierbar 92
- Färbung 168
- Fibonacci-Funktion 128
- Folge
 - leere 34
 - Markierungs- 245
- Folgerung, semantische 93, 107
- folgt semantisch 93, 107
- Formel
 - geschlossene 103
 - prädikatenlogische 88, 101
- freie Variable 103
- Funktion 38

- charakteristische 40
- konstante 40
- n-stellige 40
- Funktionsform 61
- G**
- gebundene Variable 103
- gegenseitiger Ausschluss 247, 256
- Generalisierung 219
- geordnetes Paar 30
- gerichteter Baum 157
- gerichteter Graph 137
- geschachtelt 197
- gewurzelter Baum 185
- gleichbedeutend 76
- Gleichheit 113
- Gleichheitsprädikat 113
- Grad
 - Ausgangs- 141
 - eines Graphen 141
 - eines Knotens 141
 - Eingangs- 141
- Grammatik
 - kontextfreie 163, 181, 182, 183, 200
 - mehrdeutige 187
- Graph 135, 288
 - Abhängigkeits- 168, 268
 - Ablauf- 173
 - Aufruf- 175
 - azyklischer 146
 - gerichteter 137
 - induzierter Teil- 140
 - Markierungs- 247
 - Multi- 143
 - planarer 167
 - Programmablauf- 173
 - Teil- 140, 147, 153, 165
 - ungerichteter 139
- Grundterm 60
- H**
- Halbordnung 37, 119, 122
 - strenge 37, 114
- Hamilton
 - Kreis 150
 - Weg 150
- Hilfskonstruktor 75
- HTML 191
- I**
- Idempotenz 95
- Identitätsfunktion 40
- Implikation 88, 120, 121
- indirekter Beweis 117, 123
- Induktion 125
- Induktionsanfang 126
- Induktionsschritt 126
- Induktionsvoraussetzung 126
- induktiv 125
- induzierter Teilgraph 140
- Infixform 61, 64, 189
- injektiv 40
- Interpretation 89, 105
 - erfüllende 107
 - syntaktisch passende 105
- irreflexiv 36, 114
- IST-Hierarchie 272
- IST-Relation 211, 219
- J**
- Junktor 88
- K**
- Kanten 137
 - gewicht 254
 - markierung 142
 - unabhängige 165
- Kapazität 253
- Kardinalität 28, 209, 217, 267
- kartesische Produkt 30
- Keller 74
 - Prinzip 74
- Kennzeichen 195
- Kennzeichenkomponente 33
- Kettenschluss 121
- KFG 183
- k-Klausel 96
- Klammer 195, 199
- Klammergrammatik 183
- klammern, vollständig 62
- Klasse 216

Klassendiagramm 162, 202, 215, 289
 Klausel 96
 negative 96
 positive 96
 KNF 97
 Knoten 137
 -grad 141
 -markierung 142
 -markierung, konfliktfreie 167
 Knotenmarkierung
 konfliktfreie 167
 Kommutativität 95
 Komplement 95
 Komposition 219
 Konflikt 247, 270
 konfliktfreie Knotenmarkierung, 167
 Königsberger Brückenproblem 144
 Konjunktion 88, 120
 konjunktive Normalform 97
 Konklusion 117
 konkrete Ausprägung 203, 206, 211, 216
 konsistente Umbenennung 103
 Konstante 40, 59
 Konstruktor 75
 Konsumenten-Prozess 254
 kontextfreie Grammatik 163, 181, 182,
 183, 200, 289
 korrekter Term 60, 88, 101
 Kreis 146
 kritisch
 Abschnitt 247
 247
 Pfad 170, 172
L
 Laufzeitkeller 74
 lebendig 251
 schwach 251
 Leser-Prozess 254
 Leser-Schreiber-System 254
 LIFO 74
 links-abwärts 63
 -Durchlauf 64
 linksassoziativ 62, 191
 Literal 89

Logik 87, 288
 logisch äquivalent 94, 108

M

Marke 245
 Markierung 245
 Anfangs- 247
 Kanten- 142
 Knoten- 142
 Nachfolge- 246
 Markierungsfolge 245
 Markierungsgraph 247
 Matching 165
 Mealy-Automat 238
 mehrdeutig 187
 Menge 27, 287
 Teil- 22, 28
 model checking 21
 Modell 18, 19, 287
 Modellierung 16, 26, 44, 70, 87, 99,
 100, 113, 136, 215, 227, 263, 287
 duale 172
 Modus Ponens 119
 Moore-Automat 238
 Multigraph 143
 Multimenge 41
 multiplicity 217

N

Nachbereich 245, 251
 Nachfolgemarkierung 246
 Nachfolgezustand 232
 Name-Wert-Paare 196
 nebenläufig 227, 243, 244, 270
 Negation 88, 95
 Negationsnormalform 95, 109
 Neutrale Elemente 95
 nicht-deterministisch 246
 endlicher Automat 235
 Nichtterminal 183
 -symbol 183, 200
 NNF 95, 109
 node 137
 Normalform 76, 97
 disjunktive 98

konjunktive 97
pränexe 110
n-stellig 35, 40, 59, 61, 63
n-stellige Funktion 40
n-stellige Relation 35, 206

O

Oberklasse 219
Operand 59
operational 23
Operator 59
Operatorsymbol 59
Ordnung 122
 lineare 37
 partielle 37
 strenge 37, 122
 totale 37, 114, 122
orientierbar 155

P

Paar, geordnetes 30
paarweise Zuordnung 164
Petri-Netz 227, 244, 245, 269, 276, 290
 lebendiges 251
 schwach lebendig 251
Pfad, kritischer 170, 172
planar 167
planarer Graph 167
Postfixform 61, 64
Potenzmenge 29
Prädikat 41
Prädikatenlogik 100, 116, 267, 274, 287
 erster Stufe 102
 Semantik der 104
 Syntax 100
prädikatenlogische Formel 88, 101
Präfix 110
Präfixform 61, 64, 188
Prämisse 117
pränexe Normalform 110
Präzedenz 62, 191
Präzedenzregeln 88
Primformeln 101
Primzahlen 118
Produkt, kartesisches 30

Produktion 183, 200
Produzenten-Prozess 254
Programmablaufgraph 173
Projektion 75
Prozess 247
 Konsumenten- 254
 Leser- 254
 Produzenten- 254
 Schreiber- 254
Puffer 254
Pythagoras 117

Q

q.e.d. 117
Quantifizierung 108, 109
Quantor 102, 123
 bindet 102
 -elimination 108
 -tausch 108
 -wechsel 108
 -zusammenfassung 108
Quasiordnung 37

R

rechtsassoziativ 62
reflexiv 36, 114
regelbasiert 181
regulärer Ausdruck 201, 230
Relation 202, 245, 288
 2-stellig 36, 38
 IST- 211
 n-stellige 35, 206
 symmetrische 125
Rundweg 144

S

Satz 117
Schachtelung 197, 200
schalten 246
Schaltfolgen 248
Schaltregel 246
scheduling 169
Schleife 137
Schlinge 137
Schluss, Ketten- 121
Schlüsselattribut 206

- Schlussregel, logische 119
 Schnittknoten 155
 Schreiber-Prozess 254
 Semantik
 der Aussagenlogik 89
 der Prädikatenlogik 104
 SGML 194, 289
 sicher 250
 Signatur 38, 59, 71, 104, 105, 188
 SKNF 112
 Skolemisierung 113
 Skolem-Normalform 112
 Sorte 59, 71
 Spannbaum 153
 Spezifikation, algebraische 80
 Sprache 185, 248
 akzeptierte 233, 236
 Standardized Generalized Markup Language 194, 289
 stark zusammenhängend 147
 starke Zusammenhangskomponente 147
 Startsymbol 183
 Statecharts 215, 241, 289
 Stelle 245
 strenge Halbordnung 37, 114
 strenge Ordnung 37
 Strukturbaum 164
 Strukturbeschreibungen 59
 Substitution 64
 einfache 65
 leere 67
 mehrfache 66
 surjektiv 39
 Symbole 183
 symmetrisch 36, 114, 119, 125
 Synchronisation 247, 255
 syntaktisch passende Interpretation 105
 Syntax, abstrakte 164
- T**
 Tag 195
 Anfangs- 195
 End- 195
 field 33
 -Name 195
- Tautologie 92
 tautologisch 92
 Teil
 -automat 241
 -baum 157, 199
 -graph 140, 147, 153, 165
 -graph, induzierter 140
 -menge 22, 28
 Term 57, 287
 -Algebra 71
 gleichbedeutend 76
 in Normalform 76
 korrekter 60
 undefinierter 76
 Terminal
 -symbol 183, 200
 total 36, 37, 39
 totale Ordnung 114
 Transition 245
 lebendige 251
 transitiv 36, 37, 114
 Typ 214
- U**
 Übergang 241
 Übergangsfunktion 232, 233, 235
 Umbenennung, konsistente 103
 umfasst 67
 Umformungsgesetze 108, 122
 Umformungsregel 123
 UML 181, 202, 215, 241, 289
 unabhängige Kanten 165
 undefiniert 76
 unerfüllbar 92
 ungerichteter Baum 152
 ungerichteter Graph 139
 Unified Modeling Language 202, 215
 Unifikation 65, 68
 Unifikator 68
 allgemeinster 69
 unifizierbar 68
 Unterklassen 219
- V**
 Variable 60, 64

freie 103
gebundene 103
vereinigter Wertebereich 32
Vereinigung 28
disjunkte 33
Verklemmung 252
vertex 137
Verwandtschaft 115
Vokabular 183
vollständig 158
Automat 233
klammern 62
Voraussetzung 117
Vorbereich 245, 251

W

W3C 194
Wahrheitstafel 91
Wahrheitswert 89
Weg 144, 145
Wegeprobleme 144
Wertebereich 25, 38, 123, 266, 272, 287
vereinigter 32
Widerspruch 118
widerspruchsvoll 92
Wirkungsbereich 103
wohlgeformt 197
World Wide Web 194
Consortium 194

Wort 230
Wurzel 157
WWW 194

X

XMI 215
XML 191, 194, 289
XML Metadata Interchange 215
XML-Datei 197
XML-Parser 201

Z

Zahl, chromatische 168
Zeichen 230
-folgen 230
Zuordnung, paarweise 164
Zuordnungsproblem 164
zusammenhängend 147
stark 147
Zusammenhangskomponente 147
starke 147
Zustand 232
Anfangs- 232
End- 232
Nachfolge- 232
zusammengesetzter 242
Zyklus 146

Tabelle 0.1 Auf plus.hanser-fachbuch.de bereitgestellte Folien zur Vorlesung „Modellierung“

| Titel | Foliennummer |
|---|---------------------|
| 0. Modellierung | 100 |
| 1. Einführung | |
| <i>Begründung der Vorlesung</i> | 101 |
| <i>Ziele</i> | 102 |
| <i>Durchführung</i> | 103 |
| <i>Inhalt</i> | 104 |
| <i>Literaturhinweise</i> | 105 |
| <i>Bezüge zu anderen Vorlesungen</i> | 106 |
| <i>Elektronisches Skript: Startseite</i> | 107 |
| <i>Elektronisches Skript: Organisation</i> | 108 |
| <i>Elektronisches Skript: Regeln</i> | 109 |
| <i>Elektronisches Skript: kommentierte Folien</i> | 111 |
| <i>Beispiel: Die Flussüberquerung</i> | 112 |
| <i>Beispiel: Die Flussüberquerung</i> | 112a |
| <i>Modellierung der Flussüberquerung</i> | 113 |
| <i>Diskussion des Modellierungsbeispiels</i> | 114 |
| <i>Modellierungsbeispiel: Getränkeautomat</i> | 114a |
| <i>Allgemeiner Modellbegriff</i> | 115 |
| <i>Modell: Buslinienplan</i> | 115a |
| <i>Modell: Busfahrplan</i> | 115b |
| <i>Modellbegriff im allgemeinen Lexikon</i> | 115e |
| <i>Modellbegriff im Lexikon der Informatik</i> | 115f |
| <i>Zweck des Modells</i> | 116 |
| <i>Arbeiten mit dem Modell</i> | 117 |
| <i>Bezug zwischen Original und Modell</i> | 118 |
| <i>Modellierte Aspekte</i> | 119 |
| <i>Deklarative oder operationale Beschreibung</i> | 120 |
| 2. Modellierung mit Wertebereichen | 201 |
| <i>Übersicht über Begriffe</i> | 202 |
| <i>Einführendes Beispiel</i> | 203 |
| <i>Wertebereiche für das Beispiel</i> | 204 |
| <i>2.1 Mengen</i> | 205 |
| <i>Eigenschaften von Mengen</i> | 205a |
| <i>Russels Paradoxon</i> | 205r |

| Titel | Foliennummer |
|---|--------------|
| <i>Mengenoperationen</i> | 206 |
| <i>2.2 Potenzmengen</i> | 207 |
| <i>Modellierung mit Potenzmengen</i> | 207a |
| <i>2.3 Kartesische Produkte</i> | 208 |
| <i>2.4 Disjunkte Vereinigung</i> | 208a |
| <i>2.5 Folgen</i> | 208b |
| <i>2.6 Relationen</i> | 209 |
| <i>Kardinalität, Schreibweisen</i> | 209a |
| <i>Eigenschaften 2-stelliger Relationen</i> | 210 |
| <i>Beispiele für Eigenschaften 2-stelliger Relationen</i> | 210a |
| <i>Ordnungsrelationen</i> | 210c |
| <i>Beispiele für Ordnungsrelationen</i> | 210d |
| <i>2.7 Funktionen</i> | 211 |
| <i>Beispiele für Funktionen</i> | 211b |
| <i>Eigenschaften von Funktionen</i> | 212 |
| <i>Spezielle Funktionen</i> | 213 |
| <i>Funktionen zur Modellierung von mehrfachen Vorkommen</i> | 213a |
| <i>Funktionen auf Indexmengen</i> | 213b |
| <i>Hinweise zum Modellieren mit Wertebereichen</i> | 214a |
| <i>Wertebereiche zur Modellierung des Getränkeautomaten</i> | 214b |
| <i>2x Beweise verstehen und konstruieren</i> | 251 |
| <i>Beispiel 1</i> | 252 |
| <i>Gegenbeispiel</i> | 253 |
| <i>Beispiel 2</i> | 254 |
| <i>Eigenschaften von Beweisen</i> | 255 |
| <i>Form von Satz und Beweis</i> | 256 |
| <i>Beweisstruktur Fallunterscheidung</i> | 257 |
| <i>Implikation als Behauptung</i> | 258 |
| <i>Beweisstruktur ausfüllen</i> | 259 |
| <i>Konstruktionshilfen am Beispiel für Beweis 2x.3</i> | 259b |
| <i>Konstruktionshilfen am Beispiel für Beweis 2x.3</i> | 259c |
| <i>Konstruktionshilfen am Beispiel für Beweis 2x.3</i> | 259d |
| <i>Konstruktionshilfen am Beispiel für Beweis 2x.3</i> | 259e |
| <i>Konstruktionshilfen am Beispiel für Beweis 2x.3</i> | 259f |
| <i>Konstruktionshilfen am Beispiel für Beweis 2x.3</i> | 259g |
| <i>Konstruktionshilfen am Beispiel für Beweis 2x.3</i> | 259h |