

Mehrwertige Logiken

Many-valued logics

Fabian Koßmann

Seminar

Betreuer: Prof. Dr. Claudia Schon

Trier, 26.01.2024

Kurzfassung

Zunächst werden in dieser Seminararbeit die Grundelemente einer n-wertigen Logik vorgestellt. Im Anschluss erfolgt die Erläuterung einer mehrwertigen Logik anhand eines konkreten Beispiels. Dabei wird die dreiwertige Lukasiewicz Logik L_3 als Beispiel für eine n-wertige Logik eingeführt und im weiteren Verlauf um prädikatenlogische Aspekte erweitert. Abschließend wird die Vienna Development Method Specification Language VDM-SL als Anwendungsbeispiel betrachtet. Zum Abschluss der Arbeit wird die (VDM-SL) auf ein konkretes Beispiel angewandt, um einen Einblick in die praktische Anwendung zu geben.

Inhaltsverzeichnis

1	Die Geschichte und Idee hinter mehrwertigen Logiken	1
2	n-wertige Logiken	2
2.1	Syntax	2
2.1.1	Festlegen der Bezeichner für die Wahrheitswerte	2
2.1.2	Festlegen der Bezeichner für Operatoren	2
2.1.3	Festlegen der Bezeichner für aussagenlogische Variablen	2
2.1.4	Vokabular	2
2.1.5	Menge der Formeln	3
2.2	Semantik	3
2.2.1	Belegung	3
2.2.2	Festlegung der Bedeutung der Operatoren	3
2.2.3	F-Algebra	3
2.2.4	Bewertungsfunktion	3
2.2.5	Menge von designierten Wahrheitswerten	3
2.3	Weitere Bestandteile einer n-wertigen Logik	4
2.3.1	Erfüllbarkeit	4
2.3.2	Unerfüllbarkeit	4
2.3.3	Tautologie	4
2.3.4	Kalküle	4
3	Beispiel für eine n-Wertige Logik	5
3.1	Einführung der L_3 -Logik auf Basis der Aussagenlogik	5
3.1.1	Syntax	5
3.1.2	Semantik	6
3.2	Erweiterung der L_3 -Logik zur Prädikatenlogik	11
3.2.1	Syntax	11
3.2.2	Festlegung der Bezeichner für die Wahrheitswerte	11
3.2.3	Semantik	13
3.3	Weitere Bestandteile der L_3 -Logik	16
3.3.1	Erfüllbarkeit, Unerfüllbarkeit und Tautologie in der L_3 -Logik	16
3.3.2	L_3 -Struktur	17
3.3.3	Folgerung in der L_3 -Logik	17

3.3.4 Kalküle.....	17
4 Anwendungsbeispiel	18
4.1 Vienna Development Method (VDM)	18
4.1.1 Korrespondenz zwischen der L_3 -Logik und der VDM-SL.....	18
4.1.2 Beispiel für die VDM-SL	20
5 Zusammenfassung und Ausblick	21
Literaturverzeichnis	22
Selbstständigkeitserklärung	23

Tabellenverzeichnis

3.1	Wahrheitstabelle der schwachen Negation	7
3.2	Wahrheitstabelle der starken Negation	7
3.3	Wahrheitstabelle des Und-Operator	8
3.4	Wahrheitstabelle des Oder-Operator	8
3.5	Wahrheitstabelle der schwachen Implikation	9
3.6	Wahrheitstabelle der starken Implikation	9
3.7	Wahrheitstabelle der schwachen Implikation	10
3.8	Wahrheitstabelle der starken Implikation	10
4.1	Wahrheitswerte in der VDM-SL ([Sch04a], S. 83)	18
4.2	Operatoren in der VDM-SL ([Sch04a], S. 83)	19
4.3	Prädikate in der VDM-SL ([Sch04a], S. 83)	19

Die Geschichte und Idee hinter mehrwertigen Logiken

Im nachfolgenden Abschnitt wird ein kurzer Überblick über die Geschichte und die grundlegende Idee hinter mehrwertigen Logiken gegeben.

Zunächst hatte die mehrwertige Logik vor allem philosophische Bedeutung. Mathematiker und Philosophen setzten sich mit der Frage auseinander, welchen Wahrheitswert nicht eindeutig bestimmbare Aussagen erhalten. In der klassischen Aussagenlogik werden lediglich zwei Wahrheitswerte verwendet. Dies hat jedoch zur Folge, dass viele alltägliche Probleme mit ihr nicht ausgedrückt werden können. Beispielsweise lassen sich nicht eindeutig bestimmbare Aussagen nicht ausdrücken. ([Sch04a], S. 20)

Ein Beispiel hierfür wäre die Aussage „Morgen wird es regnen“. Diese Aussage lässt sich nicht eindeutig als wahr oder falsch bestimmen. Daher kann sie nicht durch eine zweiwertige Logik ausgedrückt werden. Der Grund liegt darin, dass der Zeitpunkt, an dem man feststellen kann, ob es morgen regnet, in der Zukunft liegt. Durch die Einführung eines dritten Wahrheitswerts für eine solche nicht eindeutige Aussage könnte dieses Problem jedoch gelöst werden.

Im Folgenden werden einige bedeutende Persönlichkeiten aus der Wissenschaft kurz erwähnt, die die mehrwertige Logik maßgeblich geprägt haben.

Im Jahr 1920 veröffentlichte Jan Lukasiewicz einen Artikel, der durch die zuvor erläuterten philosophischen Überlegungen motiviert war. Diese Überlegungen betrafen die Frage nach dem Wahrheitswert einer Aussage, die erst in der Zukunft als wahr oder falsch bestimmt werden kann. ([Sch04a], S. 20)

Ein weiterer bedeutender Name in der mehrwertigen Logik ist Emil Post. Im Jahr 1921 veröffentlichte er eine Arbeit über mehrwertige Logiken. Seine Motivation lag in der Untersuchung einer bestimmten Klasse von Algebren mithilfe mehrwertiger Logik. ([Sch04a], S. 20)

Ein weiterer wichtiger Beitrag zur Entwicklung mehrwertiger Logiken stammt von Stephen Kleene. Dieser entwickelte eine dreiwertige Logik, um den Umgang mit partiell rekursiven Funktionen zu ermöglichen. ([Sch04a], S. 20)

n-wertige Logiken

Im Folgenden werden die Bestandteile eingeführt, die für die Definition einer mehrwertigen Logik benötigt werden.

2.1 Syntax

Im folgenden Abschnitt wird die Syntax, also der Aufbau von Formeln in der allgemeinen Aussagenlogik, erweitert. Dadurch wird es möglich, die Syntax auf eine beliebige endliche Menge von Wahrheitswerten anwendbar zu machen.

2.1.1 Festlegen der Bezeichner für die Wahrheitswerte

Zu Beginn wird die Anzahl der Wahrheitswerte m festgelegt, wobei $m \geq 2$ gelten soll. Außerdem wird eine Menge M von Bezeichnern für sämtliche Wahrheitswerte in der Logik eingeführt. ([Sch04a], S. 23)

2.1.2 Festlegen der Bezeichner für Operatoren

In der mehrwertigen Logik besteht die Möglichkeit, eine Vielzahl beliebiger Operatoren zu verwenden. Die Bezeichner für die Operatoren werden durch die Menge F festgelegt. Zudem ist für jedes Operatorsymbol eine Stelligkeit n anzugeben. In der normalen Aussagenlogik umfasst die Menge F beispielsweise die Symbole $\neg, \wedge, \vee, \implies, \Leftrightarrow$. ([Sch04a], S. 23f.)

2.1.3 Festlegen der Bezeichner für aussagenlogische Variablen

Die Bezeichner für die aussagenlogischen Variablen, auch als nullstellige Prädikate bezeichnet, werden durch die Menge P festgelegt. Diese Variablen werden später mit den entsprechenden Wahrheitswerten belegt. ([Sch04a], S. 24)

2.1.4 Vokabular

Das Vokabular umfasst alle Symbole, die in einer Logik verwendet werden dürfen. Es setzt sich aus der Menge der Operatoren F , der Menge der aussagenlogischen Variablen P sowie anderen Symbolen wie Klammern zusammen. ([Sch04b], S. 9)

2.1.5 Menge der Formeln

Die Menge der Formeln $Fml(V)$ umfasst sämtliche möglichen Formeln der Logik und wird üblicherweise induktiv definiert. Dabei werden grundlegende Formeln festgelegt, und es wird beschrieben, wie durch die bereits existierenden Formeln neue gebildet werden können. ([Sch04b], S. 10)

2.2 Semantik

Im folgenden Abschnitt wird die Semantik der allgemeinen Aussagenlogik erweitert, um sie auf eine beliebige endliche Menge von Wahrheitswerten anwendbar zu machen.

2.2.1 Belegung

Eine Belegung μ für eine Vokabular V ist eine Funktion $\mu : P \rightarrow M$, die jeder aussagenlogischen Variablen einen Wert aus M zuweist. ([Sch04a], S. 24)

2.2.2 Festlegung der Bedeutung der Operatoren

Es muss die Semantik der Operatorsymbole festgelegt werden. Jedem Operatorbezeichner f aus der Menge F mit der Stelligkeit n wird dafür eine Funktion $f^M : M^n \rightarrow M$ zugeordnet. Diese Funktion bildet von der Menge der Wahrheitswerte in die Menge der Wahrheitswerte ab. Für ein n -stelliges Operatorsymbol wird diesen also eine Funktion zugewiesen die n Wahrheitswerten einen Wahrheitswert zuweist. ([Sch04a], S. 24)

2.2.3 F-Algebra

Die Struktur aus Operatoren und deren Bedeutungen kann auch wie folgt dargestellt werden: $\mathbf{M} = (M, f^M : f \in F)$ und wird auch als Matrix einer Logik oder F-Algebra bezeichnet. ([Sch04a], S. 24)

2.2.4 Bewertungsfunktion

Die Bewertungsfunktion $\mu_M : Fml(V) \rightarrow M$, weist jeder Formel t aus $Fml(V)$ einen Wahrheitswert aus M zu. In der Regel ist die Funktion über die F-Algebra sowie der Belegung μ definiert. Der Wahrheitswert einer Formel wird auch mit $\mu_M(t)$ bezeichnet. ([Sch04a], S. 24)

2.2.5 Menge von designierten Wahrheitswerten

Es wird eine nichtleere Teilmenge D aus der Menge der Bezeichner für die Wahrheitswerte definiert. Diese Teilmenge umfasst alle Bezeichner für Wahrheitswerte, die als wahr gelten. Diese Definition ist notwendig, um Erfüllbarkeit, Unerfüllbarkeit und Tautologie zu beschreiben. ([Sch04a], S. 24)

2.3 Weitere Bestandteile einer n -wertigen Logik

Im Folgenden werden zusätzliche Bestandteile einer n -wertigen Logik erläutert.

2.3.1 Erfüllbarkeit

Mittels der Menge der designierten Werte D kann nun die Erfüllbarkeit definiert werden. Eine Formel t ist genau dann erfüllbar, wenn für mindestens eine Belegung μ gilt, dass $\mu_M(t) \in D$ ist. ([Sch04a], S. 24)

2.3.2 Unerfüllbarkeit

Über die Menge der designierten Werte D lässt sich ebenfalls die Unerfüllbarkeit definieren. Eine Formel t ist genau dann unerfüllbar, wenn für keine Belegung μ gilt, dass $\mu_M(t) \in D$ ist. ([Sch04a], S. 24)

2.3.3 Tautologie

Ebenfalls lässt sich mit der Menge der designierten Werte D die Tautologie definieren. Eine Formel t ist genau dann eine Tautologie, wenn für alle Belegungen μ gilt, dass $\mu_M(t) \in D$ ist. ([Sch04a], S. 24)

2.3.4 Kalküle

Für n -wertige Logiken werden üblicherweise Tableauealküle zum logischen Schließen verwendet. Dabei ist es bedeutend, dass für jede endliche n -wertige Logik mit einer endlichen Menge M von Wahrheitswerten, einer Menge D von designierten Wahrheitswerten und einer Liste von Wahrheitstabellen für beliebige aussagenlogische Operatoren F ein vollständiger und korrekter Tableauealkül konstruiert werden kann. ([Sch04a], S. 52) In dieser Arbeit wird jedoch nicht näher auf Tableauealküle eingegangen. Sie werden hier lediglich kurz erwähnt, da Kalküle von essentieller Bedeutung für den Nutzen einer Logik sind. Eine ausführliche Darlegung des Tableauealküls, sowie die Bedeutung von Vollständigkeit und Korrektheit in diesem Kontext, sind jedoch nicht Bestandteil dieser Arbeit. Eine umfassende Erklärung des Tableauealküls findet sich in [Sch04a] auf den Seiten 52-58.

Beispiel für eine n-Wertige Logik

Im Folgenden wird die dreiwertige Łukasiewicz-Logik L_3 als Beispiel für eine mehrwertige Logik vorgestellt. Dabei werden die zuvor aufgeführten Elemente einer n-wertigen Logik genutzt, um die L_3 -Logik zu definieren. Zunächst erfolgt die Definition auf Grundlage der Ausgangslogik. Anschließend wird die L_3 um Prädikatenlogik erweitert, um auch Prädikate, Variablen, Konstanten, Funktionen und Quantoren zuzulassen.

3.1 Einführung der L_3 -Logik auf Basis der Aussagenlogik

In diesem Abschnitt wird die L_3 -Logik nun auf Basis der allgemeinen Aussagenlogik unter Verwendung der zuvor beschriebenen Bestandteile einer n-wertigen Logik definiert.

3.1.1 Syntax

Im ersten Schritt erfolgt die Definition der Syntax. Diese beschreibt, wie Formeln in der L_3 -Logik aufgebaut sind.

Festlegen der Bezeichner für die Wahrheitswerte

Im Folgenden werden die Bezeichner für die Wahrheitswerte der L_3 -Logik definiert. Diese bestehen aus der Menge $M = \{0, u, 1\}$. Im Grunde genommen handelt es sich dabei um die Bezeichner der Wahrheitswerte der Ausgangslogik, die um den Wahrheitswert u erweitert werden. Dieser beschreibt einen unbekannten, nicht bestimmbaren Wahrheitswert. ([Sch04a], S. 25)

Festlegen der Bezeichner für Operatoren

Die Menge der Bezeichner für die Operatoren in der L_3 -Logik umfasst die Symbole $\neg, \sim, \wedge, \vee, \rightarrow, \subset, \longleftrightarrow, \equiv$. Die Stelligkeit der einzelnen Operatoren ist wie folgt festgelegt: $(\neg, 1), (\sim, 1), (\wedge, 2), (\vee, 2), (\rightarrow, 2), (\subset, 2), (\longleftrightarrow, 2), (\equiv, 2)$, wobei die zweite Zahl die jeweilige Stelligkeit angibt.

Ein wesentlicher Unterschied zur klassischen Logik besteht darin, dass sowohl eine starke Negation $\neg$ als auch eine schwache Negation $\sim$ vorhanden ist. Darüber hinaus werden zur Unterstützung eine starke Implikation $\rightarrow$ und eine schwache Implikation $\supset$ sowie eine starke Äquivalenz $\longleftrightarrow$ und die schwache Äquivalenz $\equiv$ eingeführt. ([Sch04a], S. 24-28)

Festlegen der Bezeichner für aussagenlogische Variablen

Es erfolgt die Definition einer Menge von Bezeichnern P für aussagenlogische Variablen bzw. nullstellige Prädikate. Die Namensgebung der Bezeichner hängt stark von den jeweiligen Situationen ab. ([Sch04a], S. 24)

Vokabular

Das Vokabular bzw. die Signatur setzt sich aus der Menge aussagenlogischer Variablen und der Menge der Operatoren zusammen, sowie optional aus weiteren Symbolen wie beispielsweise Klammern. ([Sch04a], S. 24)

Menge der Formeln

Nun werden Formeln definiert, die über das Vokabular V gebildet werden können. Die Menge aller Formeln wird mit $Fml_3(V)$ bezeichnet. Die Formeln werden induktiv wie folgt definiert:

- Sei $p \in P$ eine Variable, dann ist p eine Formel.
- Seien ϕ und α Formeln, dann sind auch die folgenden Ausdrücke Formeln:

$$\begin{aligned} &\phi \wedge \alpha, \\ &\phi \vee \alpha, \\ &\neg \phi, \\ &\sim \phi, \\ &\phi \rightarrow \alpha, \\ &\phi \supset \alpha, \\ &\phi \longleftrightarrow \alpha, \\ &\phi \equiv \alpha. \end{aligned}$$

([Sch04b], S. 10-11)

3.1.2 Semantik

Im Folgenden wird die Semantik, also die Bedeutung der Formeln, in der Logik L_3 festgelegt.

Belegung

Die Belegung $\mu(p)$ in der L_3 -Logik ist eine Funktion, die jeder aussagenlogischen Variablen $p \in P$ einen der Werte 0, u oder 1 zuordnet. ([Sch04a], S. 24)

Ordnung nach Wahrheitsgehalt

Die Anordnung nach Wahrheitsgehalt bietet eine Möglichkeit, die Symbole für die Wahrheitswerte in der Menge M zu sortieren. Dies kann beispielsweise hilfreich sein, um Operatoren einfacher zu definieren. Im Folgenden wird die Ordnung nach dem Wahrheitsgehalt für die L_3 -Logik festgelegt. Die Ordnung lautet: $0 < u < 1$. ([Sch04a], S. 25)

Festlegung der Bedeutung der Operatoren

Wie oben gezeigt wurde, muss jedem Operator in der mehrwertigen Logik eine Funktion zugeordnet werden, die von M^n nach M abbildet. Im Folgenden werden diese Funktionen festgelegt. Es gibt verschiedene Möglichkeiten, die Operatoren in der L_3 -Logik zu definieren. Dies hängt in der Regel vom Anwendungsfall ab. ([Sch04a], S. 26)

Eine mögliche Definition der Operatoren sieht wie folgt aus:

Negations-Operatoren

Die schwache Negation ist ein unärer Operator. Im Folgenden wird die schwache Negation $\sim$ über eine Wahrheitstabelle definiert.

p	$\sim p$
1	0
u	1
0	1

Tabelle 3.1: Wahrheitstabelle der schwachen Negation

Wichtig hierbei ist, dass ein unbekannter Wert bei der schwachen Negation nicht unbekannt bleibt, sondern den Wert 1 annimmt. ([Sch04a], S. 26)

Die starke Negation ist ebenfalls ein unärer Operator. Im Folgenden wird die starke Negation $\neg$ über eine Wahrheitstabelle festgelegt.

p	$\neg p$
1	0
u	u
0	1

Tabelle 3.2: Wahrheitstabelle der starken Negation

Insbesondere wichtig hierbei ist, dass ein unbekannter Wert bei der starken Negation unbekannt bleibt. ([Sch04a], S. 26)

Die starke Negation hebt sich weiterhin auf, daher hat $\neg\neg$ den gleichen Wahrheitswert wie zuvor. Für die schwache Negation gilt das jedoch nicht, wie aus der obigen Tabelle ersichtlich ist. Ebenso sind die starken und schwachen Negationen nicht kommutativ, es gilt also $\neg \sim \neq \sim \neg$. ([Sch04a], S. 26)

Und-Operator

Im Folgenden erfolgt die Definition des Und-Operators. Dieser binäre Operator ist, wie in vielen Programmiersprachen, definiert. Die Auswertung von „falsch $\wedge$ unbekannt“ und „unbekannt $\wedge$ falsch“ führt zu „falsch“, während die Auswertung von „unbekannt $\wedge$ unbekannt“ unverändert „unbekannt“ ergibt. Die genaue Definition des Und-Operators ist in der folgenden Wahrheitstabelle dargestellt.

$\wedge$	1	u	0
1	1	u	0
u	u	u	0
0	0	0	0

Tabelle 3.3: Wahrheitstabelle des Und-Operator

Eine weitere Möglichkeit zur Definition des Und-Operators ergibt sich über die zuvor eingeführte Ordnung nach Wahrheitsgehalt. Wenn μ eine Belegung ist, gilt folgendes: $\mu_M(a \wedge b)$ ist immer das Minimum aus $\mu_M(a)$ und $\mu_M(b)$. ([Sch04a], S. 25)

Oder-Operator

Der logische Oder-Operator ist ebenfalls ein binärer Operator und ist, wie in vielen Programmiersprachen, definiert. Die Auswertung von „wahr $\vee$ unbekannt“ und „unbekannt $\vee$ wahr“ führt zu „wahr“, während die Auswertung von „unbekannt $\vee$ unbekannt“ unverändert „unbekannt“ ist. Die genaue Definition des Oder-Operators ist in der folgenden Wahrheitstabelle dargestellt.

$\vee$	1	u	0
1	1	1	1
u	1	u	u
0	1	u	0

Tabelle 3.4: Wahrheitstabelle des Oder-Operator

Ebenfalls kann der Oder-Operator auch alternativ über die zuvor eingeführte Ordnung nach Wahrheitsgehalt definiert werden. Für eine Belegung μ gilt dann, dass $\mu_M(a \vee b)$ immer das Maximum aus $\mu_M(a)$ und $\mu_M(b)$ ist. ([Sch04a], S. 25)

Implikations-Operatoren

Im folgenden wird die Schwache Implikation eingeführt, die mithilfe der Schwachen Negation definiert wird und durch das Zeichen $\supset$ repräsentiert wird. Die Definition lautet wie folgt: Die Aussage „A impliziert B“ ist definiert als $A \supset B := \sim A \vee B$. Diese Definition ähnelt der in der Aussagenlogik. ([Sch04a], S. 27) Eine alternative Definition kann mithilfe der folgenden Wahrheitstabelle erfolgen.

$A \supset B$	
$B \downarrow A \rightarrow$	1 u 0
1	1 1 1
u	u 1 1
0	0 1 1

Tabelle 3.5: Wahrheitstabelle der schwachen Implikation

Die Starke Implikation ist analog zur Definition in der Aussagenlogik gestaltet und wird durch das Symbol $\rightarrow$ repräsentiert. Die Definition lautet wie folgt: Die Aussage „A impliziert B“ ist definiert als $A \rightarrow B := \neg A \vee B$. ([Sch04a], S. 28) Eine alternative Definition kann ebenfalls mithilfe der folgenden Wahrheitstabelle erfolgen.

$A \rightarrow B$	
$B \downarrow A \rightarrow$	1 u 0
1	1 u 1
u	u u 1
0	0 u 1

Tabelle 3.6: Wahrheitstabelle der starken Implikation

Äquivalenz-Operatoren

Im folgenden Abschnitt wird die schwache Äquivalenz definiert. Die Schreibweise dieser Äquivalenz lautet $A \equiv B$. Die Definition ähnelt ebenfalls der Definition in der Aussagenlogik und lautet wie folgt: $A \equiv B := (A \supset B) \wedge (B \supset A)$. ([Sch04a], S. 27)

Eine alternative Definition kann mithilfe der folgenden Wahrheitstabelle erfolgen.

$A \equiv B$	
$B \downarrow A \rightarrow$	1 u 0
1	1 u 0
u	u 1 1
0	0 1 1

Tabelle 3.7: Wahrheitstabelle der schwachen Implikation

Die Starke Äquivalenz ist ebenfalls analog zur Definition in der Aussagenlogik gestaltet und wird durch das Symbol $A \leftrightarrow B$ repräsentiert. Die Starke Äquivalenz ist wie folgt definiert: $A \longleftrightarrow B := (A \rightarrow B) \wedge (B \rightarrow A)$. ([Sch04a], S. 28)
Eine alternative Definition kann ebenfalls mithilfe der folgenden Wahrheitstabelle erfolgen.

$A \longleftrightarrow B$	
$B \downarrow A \rightarrow$	1 u 0
1	1 u 0
u	u u u
0	0 u 1

Tabelle 3.8: Wahrheitstabelle der starken Implikation

F-Algebra

Die Definition der F-Algebra bleibt unverändert, wie sie in Abschnitt 2.2.3 festgelegt ist. Die Funktionsvorschriften für die einzelnen Operatoren sind über die Wahrheitstabellen dieser Operatoren in Abschnitt 3.1.2 definiert.

Bewertungsfunktion

Die Bewertungsfunktion μ_M ordnet jeder Formel $f \in Fml_3(V)$ einen Wert aus der Menge $\{0, u, 1\}$ zu. Die Funktionsvorschrift wird durch die im Abschnitt 3.1.2 definierte Belegung sowie durch die in Abschnitt 3.1.2 festgelegte F-Algebra bestimmt. ([Sch04a], S. 24)

Die Funktion μ_M wird rekursiv wie folgt definiert:

Seien $\phi \in Fml_3(V)$ und $\beta \in Fml_3(V)$. Dann gilt folgendes:

- Wenn α eine aussagenlogische Variable ist, also $\alpha \in P$:

$$\mu_M(\alpha) = \mu(\alpha)$$

- Sonst:

$$\begin{aligned}
\mu_M(\phi \wedge \alpha) &= \mu_M(\phi) \wedge \mu_M(\alpha) \\
\mu_M(\phi \vee \alpha) &= \mu_M(\phi) \vee \mu_M(\alpha) \\
\mu_M(\neg \alpha) &= \neg \mu_M(\alpha) \\
\mu_M(\sim \alpha) &= \sim \mu_M(\alpha) \\
\mu_M(\phi \rightarrow \alpha) &= \mu_M(\phi) \rightarrow \mu_M(\alpha) \\
\mu_M(\phi \subset \alpha) &= \mu_M(\phi) \subset \mu_M(\alpha) \\
\mu_M(\phi \longleftrightarrow \alpha) &= \mu_M(\phi) \longleftrightarrow \mu_M(\alpha) \\
\mu_M(\phi \equiv \alpha) &= \mu_M(\phi) \equiv \mu_M(\alpha).
\end{aligned}$$

([Sch04b], S. 15)

Menge von designierten Wahrheitswerten

Die Menge der designierten Wahrheitswerte in der L_3 -Logik entspricht derjenigen in der Aussagenlogik und ist durch $D = \{1\}$ festgelegt. Das bedeutet, dass in der L_3 -Logik nur das Symbol 1 als wahr betrachtet wird. ([Sch04a], S. 25)

3.2 Erweiterung der L_3 -Logik zur Prädikatenlogik

In diesem Abschnitt wird die im Abschnitt 3.1 definierte L_3 -Logik auf Basis der Aussagenlogik dahingehend erweitert, dass nun auch prädikatenlogische Elemente wie Konstanten, Variablen, Funktionen, Prädikate und Quantoren verwendet werden können.

3.2.1 Syntax

In diesem Abschnitt erfolgt eine Erweiterung der Syntax. Hierfür werden Funktionssymbole, Variablensymbole und Konstantensymbole eingeführt und daraus Terme gebildet. Anschließend werden mittels Prädikatsymbolen, Termen und Operatorsymbolen Formeln konstruiert.

3.2.2 Festlegung der Bezeichner für die Wahrheitswerte

Die Menge der Wahrheitswerte bleibt unverändert, wie bereits im Abschnitt 3.1.1 definiert.

Festlegen der Bezeichner für die Funktionssymbole

Wie in der Prädikatenlogik erster Stufe, muss die Menge F_{func} von Funktionssymbolen definiert werden. Zusätzlich muss jedem Funktionssymbol eine Stelligkeit m zugeordnet werden. ([Sch04b], S. 10)

Festlegen der Bezeichner für die Prädikate

Im Gegensatz zu den Atomen bzw. aussagenlogischen Variablen, die unter 3.1.1 definiert wurden, werden nun, genau wie in der Prädikatenlogik erster Stufe, Prädikate (Relationen) verwendet. Es ist eine Menge P zu definieren, die die prädikatenlogischen Symbole enthält, und für jedes prädikatenlogische Symbol ist die Stelligkeit n festzulegen. ([Sch04b], S. 9)

Festlegen der Bezeichner für die Konstanten

Ebenso besteht die Möglichkeit, Konstanten zu verwenden, wie in der Prädikatenlogik erster Stufe. Die Symbole der Konstanten werden durch die Menge C definiert. ([Sch04b], S. 9)

Festlegen der Bezeichner für die Variablen

Es besteht ebenfalls die Möglichkeit, Variablen einzuführen, genau wie in der Prädikatenlogik erster Stufe. Die Symbole für die Variablen werden durch die Menge Va repräsentiert. ([Sch04b], S. 9)

Menge der Terme

Analog zur Prädikatenlogik erster Stufe, müssen Terme gebildet werden, um die Menge der Formeln zu definieren. Die Menge aller Terme T wird induktiv wie folgt definiert:

- Variablensymbole sind Terme.
- Konstantensymbole sind Terme.
- Sei $func \in F_{\text{func}}$ ein Funktionssymbol mit der Stelligkeit m , und seien $t_1, \dots, t_m$ Terme. Dann ist $func(t_1, \dots, t_m)$ ebenfalls ein Term.

([Sch04b], S. 10)

Menge der Operatoren

Die Menge der Operatorsymbole F wird um die beiden Operatorsymbole ($\forall$) und ($\exists$) erweitert. Die aktualisierte Menge der Operatorsymbole lautet somit: $F = \neg, \sim, \wedge, \vee, \rightarrow, \subset, \longleftrightarrow, \equiv, \forall, \exists$. Die Stelligkeit der einzelnen Operatoren ist wie folgt festgelegt: $(\neg, 1), (\sim, 1), (\wedge, 2), (\vee, 2), (\rightarrow, 2), (\subset, 2), (\longleftrightarrow, 2), (\equiv, 2), (\forall, 1), (\exists, 1)$. Die zweite Zahl gibt erneut die jeweilige Stelligkeit an. ([Sch04b], S. 9)

Vokabular

Das Vokabular bzw. die Signatur besteht nun aus der Menge der Terme T , der Menge der Prädikatensymbole P und der Menge der Operatorsymbole P , optional erweitert durch weitere Symbole, wie beispielsweise Klammern. ([Sch04b], S. 9)

Menge der Formeln

Die Bildung von Formeln erfolgt durch die Verwendung von Prädikatsymbolen, Termen und Operatorsymbolen aus dem Vokabular V . Die Menge aller Formeln wird mit $Fml_3(V)$ bezeichnet, und ihre Definition erfolgt induktiv. Sei $p \in P$ ein Prädikatsymbol mit der Stelligkeit n , und seien $t_1, \dots, t_n$ Terme aus der Menge T .

- Dann ist $p(t_1, \dots, t_n)$ eine Formel.
- Seien ϕ und α Formeln. Dann sind auch die folgenden Ausdrücke Formeln:

$$\begin{aligned} &\phi \wedge \alpha, \\ &\phi \vee \alpha, \\ &\neg \phi, \\ &\sim \phi, \\ &\phi \rightarrow \alpha, \\ &\phi \subset \alpha, \\ &\phi \longleftrightarrow \alpha, \\ &\phi \equiv \alpha, \\ &\forall_X \phi, \\ &\exists_X \phi. \end{aligned}$$

([Sch04b], S. 10-11)

3.2.3 Semantik

In diesem Abschnitt erfolgt eine Erweiterung der Semantik. Hierfür wird die Bedeutung der Wahrheitswerte, Funktionssymbole, Variablensymbole, Konstantensymbole, Prädikatensymbole und Operatorsymbole definiert.

Universum

Wie in der allgemeinen Prädikatenlogik erster Stufe muss das Universum, auch als Domäne bezeichnet, definiert werden. Das Universum wird durch die Menge U von beliebigen Objekten angegeben. ([Sch04b], S. 13)

Konstantenbelegung

Die Konstantenbelegung definiert die Bedeutung der Konstantensymbole. Sie wird durch die Funktion $C_b : C \rightarrow U$ repräsentiert, die jedem Konstantensymbol aus dem Vokabular V ein spezifisches Objekt aus dem Universum zuweist. ([Sch04b], S. 13)

Funktionsbelegung

Die Funktionsbelegung legt die Bedeutung der Funktionsymbole fest. Sie ist eine Abbildung F_b , die jedem Funktionssymbol $f \in F$ mit der Stelligkeit k aus dem Vokabular V eine Funktion $g : U^k \rightarrow U$ zuordnet. ([Sch04b], S. 13)

Prästruktur

Zur Vereinfachung wird eine sogenannte Prästruktur M_0 definiert. Diese legt die Bedeutung der Konstanten und Funktionen fest. Sie besteht aus einem Universum U von Elementen, einer Konstantenbelegung C_b und einer Funktionsbelegung F_b . Diese wurden bereits oben definiert. ([Sch04a], S. 25)

Variablenbelegung

Die Variablenbelegung gibt den Variablensymbolen eine Bedeutung. Sie ist eine Abbildung $V_b : V \rightarrow U$, die Variablensymbolen aus V ein Element aus dem Universum zuordnet. ([Sch04b], S. 13)

Interpretation von Termen

Die Interpretation von Termen erfolgt mittels der Interpretationsfunktion $I_t : T \rightarrow U$. Diese Funktion weist jedem Term aus T ein Symbol aus U zu. Als zusätzliche Parameter benötigt die Funktion I_t die zuvor definierte Prästruktur M_0 und eine Variablenbelegung V_b .

Die Funktion ist rekursiv wie folgt definiert: Sei M_0 eine Prästruktur und V_b eine Variablenbelegung, dann ist I_t wie folgt rekursiv definiert:

- Falls der Term t ein Variablensymbol ist:

$$I_t(M_0, V_b)(t) = V_b(t)$$

- Falls der Term t ein Konstantensymbol ist:

$$I_t(M_0, V_b)(t) = C_b(t)$$

- Falls der Term t eine Funktion $f(t'_1, \dots, t'_n)$ ist mit der Stelligkeit n :

$$I_t(M_0, V_b)(t) = F_b(f)(I_t(M_0, V_b)(t'_1), \dots, I_t(M_0, V_b)(t'_n))$$

([Sch04b], S. 12)

Belegung

Die Belegung $\mu(M_0, V_b)(p)$ ist eine Funktion, die jedem Prädikatsymbol $p \in P$ einen der Wahrheitswerte $0, u, 1$ zuweist. Da es sich nun jedoch nicht mehr um nullstellige Prädikate handelt, müssen die einzelnen Terme in den Prädikaten interpretiert werden. Die Funktion muss also jedem Prädikatsymbol und jeder möglichen Belegung der Terme in den Prädikatsymbolen einen der Wahrheitswerte $0, u, 1$ zuordnen.

Die Funktion $\mu(M_0, V_b)(p)$ ist folgendermaßen definiert:

Sei $p(t'_1, \dots, t'_n) \in P$ ein Prädikat, M_0 eine Prästruktur und V_b eine Variablenbelegung. Dann weist die Belegung μ dem Prädikat $p(I_t(M_0, V_b)(t'_1), \dots, I_t(M_0, V_b)(t'_n))$ einen der Wahrheitswerte $0, u, 1$ zu. ([Sch04a], S. 25)

Festlegung der Bedeutung der Operatoren

Die Bedeutung der zuvor unter 3.1.2 definierten Operatoren bleibt unverändert. Allerdings muss die Bedeutung des Allquantors und des Existenzquantors definiert werden.

Allquantor-Operator

Der Operator $\forall$ ist ein einstelliger Operator, der eine Variable in einer Formel α bindet. Im Folgenden wird der Allquantor definiert. Sei $\forall_x \alpha$ eine allquantifizierte Formel, dann gilt folgendes:

- $\mu_M(M_0, V_b)(\forall_x \alpha) = 1$ genau dann, wenn $\mu_M(M_0, V'_b)(\alpha) = 1$ für alle Variablenbelegungen V'_b gilt.
- $\mu_M(M_0, V_b)(\forall_x \alpha) = 0$ genau dann, wenn $\mu_M(M_0, V'_b)(\alpha) = 0$ für eine Variablenbelegung V'_b gilt.
- Ansonsten hat die Formel den Wert u bzw. ist unbekannt.

([Sch04a], S. 27) Die rekursive Auswertung der Formel α jeweils unter der Belegung V'_b sowie die Bildung der Belegungen V'_b werden im Abschnitt 3.2.3 erklärt.

Existenzquantor-Operator

Der Operator $\exists$ ist ebenfalls ein einstelliger Operator, der eine Variable in einer Formel α bindet. Im Folgenden wird der Existenzquantor definiert. Sei $\exists_x \alpha$ eine existenzquantifizierte Formel, dann gilt folgendes:

- $\mu_M(M_0, V_b)(\exists_x \alpha) = 1$ genau dann, wenn $\mu_M(M_0, V'_b)(\alpha) = 1$ für eine Variablenbelegung V'_b gilt.
- $\mu_M(M_0, V_b)(\exists_x \alpha) = 0$ genau dann, wenn $\mu_M(M_0, V'_b)(\alpha) = 0$ für alle Variablenbelegungen V'_b gilt.
- Ansonsten hat die Formel den Wert u bzw. ist unbekannt.

([Sch04a], S. 27) Die rekursive Auswertung der Formel α jeweils unter der Belegung V'_b sowie die Bildung der Belegungen V'_b wird ebenfalls im Abschnitt 3.2.3 erklärt.

F-Algebra

Die Definition der F-Algebra für die L_3 -Logik bleibt größtenteils unverändert und wurde bereits wie oben unter 3.1.2 festgelegt. Allerdings wurde die F-Algebra um die Operatoren $\forall$ und $\exists$ erweitert.

Bewertungsfunktion

Die Bewertungsfunktion μ_M bleibt weitgehend gleich zu der bereits im Abschnitt 3.1.2 definierten Bewertungsfunktion. Der Hauptunterschied besteht darin, dass in der Funktion sowohl die unter 3.2.3 definierte Prästruktur, als auch die unter 3.2.3 definierte Variablenbelegung benötigt wird. Dies liegt daran, dass für die unter 3.2.3 definierte Belegung die unter 3.2.3 definierte Interpretation der Terme

benötigt wird. Ein weiterer Unterschied liegt darin, dass die F-Algebra um die Operatoren $\forall_x$ und $\exists_x$ erweitert wurde. ([Sch04a], S. 24)

Die Funktion μ_M wird rekursiv wie folgt definiert:

Seien $\phi \in Fml_3(V)$ und $\beta \in Fml_3(V)$. Dann gilt folgendes:

- Wenn α ein Prädikat ist, also $\alpha \in P$:

$$\mu_M(M_0, V_b)(\alpha) = \mu(M_0, V_b)(\alpha)$$

- Sonst:

$$\mu_M(M_0, V_b)(\phi \wedge \alpha) = \mu_M(M_0, V_b)(\phi) \wedge \mu_M(M_0, V_b)(\alpha)$$

$$\mu_M(M_0, V_b)(\phi \vee \alpha) = \mu_M(M_0, V_b)(\phi) \vee \mu_M(M_0, V_b)(\alpha)$$

$$\mu_M(M_0, V_b)(\neg \alpha) = \neg \mu_M(M_0, V_b)(\alpha)$$

$$\mu_M(M_0, V_b)(\sim \alpha) = \sim \mu_M(M_0, V_b)(\alpha)$$

$$\mu_M(M_0, V_b)(\phi \rightarrow \alpha) = \mu_M(M_0, V_b)(\phi) \rightarrow \mu_M(M_0, V_b)(\alpha)$$

$$\mu_M(M_0, V_b)(\phi \subset \alpha) = \mu_M(M_0, V_b)(\phi) \subset \mu_M(M_0, V_b)(\alpha)$$

$$\mu_M(M_0, V_b)(\phi \longleftrightarrow \alpha) = \mu_M(M_0, V_b)(\phi) \longleftrightarrow \mu_M(M_0, V_b)(\alpha)$$

$$\mu_M(M_0, V_b)(\phi \equiv \alpha) = \mu_M(M_0, V_b)(\phi) \equiv \mu_M(M_0, V_b)(\alpha)$$

$$\mu_M(M_0, V_b)(\forall_x \alpha) = \forall_x \text{ berechne für alle } u \in U \mu_M(M_0, V'_b)(\alpha)$$

$$\mu_M(M_0, V_b)(\exists_x \alpha) = \exists_x \text{ berechne für alle } u \in U \mu_M(M_0, V'_b)(\alpha)$$

wobei V'_b wie folgt definiert ist:

$$V'_b(y) = V_b(y) \text{ genau dann, wenn } x \neq y$$

$$V'_b(y) = u \text{ genau dann, wenn } x = y$$

Hierbei ist für die Auswertung der Operatoren $\forall_x$ und $\exists_x$ die Variablenbelegung V'_b relevant. Bei der Variablenbelegung V'_b bleibt die Variablenbelegung gleich bei allen Variablen, die nicht der x -Variable des Operators entsprechen. An den Stellen, wo eine Variable der Variable x des Operators entspricht, bleibt die Variablenbelegung nicht gleich und die Variable x wird mit dem Wert u belegt. ([Sch04b], S. 15)

Menge von designierten Wahrheitswerten

Die Menge der Menge der designierten Wahrheitswerte bleibt unverändert wie bereits im Abschnitt 3.1.2 festgelegt.

3.3 Weitere Bestandteile der L_3 -Logik

3.3.1 Erfüllbarkeit, Unerfüllbarkeit und Tautologie in der L_3 -Logik

Diese sind genauso definiert wie oben unter 2.3.1, 2.3.2 und 2.3.3, daher erfordern sie keine zusätzlichen Erläuterungen. Einige interessante Eigenschaften sind jedoch

erwähnenswert. Jede L_3 -Tautologie ist auch eine L_2 -Tautologie. Die Begründung hierfür ist trivial, da die Operatoren, wie leicht überprüft werden kann, in Bezug auf die Werte null und eins mit denen der L_2 -Logik übereinstimmen. Somit ergibt sich, dass bei ausschließlicher Berücksichtigung der Wahrheitswerte null und eins in der L_3 -Logik ebenfalls der Wahrheitswert 1 für eine Tautologie resultiert. Dies gilt jedoch nicht umgekehrt, wie an der Formel $\neg A \vee A$ leicht zu erkennen ist. ([Sch04a], S. 32)

3.3.2 L_3 -Struktur

Im Folgenden wird kurz der Begriff der L_3 -Struktur definiert. Eine L_3 -Struktur $\mathbf{M} = \langle M_0, \mu_M \rangle$ zum Vokabular V besteht aus der Prästruktur zusammen mit der Bewertungsfunktion. ([Sch04a], S. 25)

3.3.3 Folgerung in der L_3 -Logik

Im Folgenden wird die Folgerung in der L_3 -Logik definiert. Eine Formel α folgt genau dann aus einer Formelmenge S , wenn für alle L_3 -Strukturen $\mathbf{M}$ zum Vokabular V gilt, dass wenn $\mu_M(\beta) = 1$ für alle $\beta \in S$ gilt, dann muss auch $\mu_M(\alpha) = 1$ sein. Die Schreibweise dafür lautet auch $S \vdash_3 A$. ([Sch04a], S. 27)

3.3.4 Kalküle

Wie bereits in 2.3.4 beschrieben, werden für n -wertige Logiken in der Regel Tableaunkalküle verwendet. In [Sch04a] (S. 52-69) ist ein Tableaunkalkül mit Schlussregeln für die L_3 -Logik zu finden.

Anwendungsbeispiel

Im Folgenden Kapitel wird ein kurzes Praxisbeispiel vorgestellt, für welches die mehrwertige Logik verwendet wird.

4.1 Vienna Development Method (VDM)

Die „Vienna Development Method“ (VDM) ist „Eine Methode zur formalen Spezifikation und Entwicklung von Computerprogrammen“ ([Sch04a], S. 81). Teil dieser Methode ist die „Vienna Development Method Specification Language“ (VDM-SL) ([Sch04a], S. 82). Im Folgenden soll die Vienna Development Method Specification Language (VDM-SL) kurz vorgestellt werden.

4.1.1 Korrespondenz zwischen der L_3 -Logik und der VDM-SL

Im diesem Abschnitt wird darauf eingegangen, wie die L_3 -Logik und die VDM-SL zueinander in Verbindung stehen. Für die Beschreibung wurde damals eine dreiwertige Logik verwendet. ([Sch04a], S. 82)

Universum

Als Universum sollen die Natürlichen Zahlen $\mathbb{N}$ dienen. Wichtig ist, dass bei der VDM-Logik immer ein Fehlerelement $*\mathbb{N}$ dem Universum hinzugefügt werden muss. Dieses dient dazu, ein Prädikat mit einem Fehler zu belegen. ([Sch04a], S. 82)

Menge der Wahrheitswerte

Die Menge der Wahrheitswerte ist wie folgt definiert:

Korrespondenz der Wahrheitswerte zwischen VDM-Logik und L_3 -Logik	
VDM-Logik	L_3 -Logik
<i>true</i>	1
$*\mathbb{B}$	<i>u</i>
<i>false</i>	0

Tabelle 4.1: Wahrheitswerte in der VDM-SL ([Sch04a], S. 83)

Der Wahrheitswert $*\mathbb{B}$ repräsentiert hierbei das Fehlerelement, das verwendet wird, um ein Prädikat mit einem Fehler zu belegen ([Sch04a], S. 82).

Operatoren

Die Menge der Operatoren in der VDM-Logik ist eine Teilmenge der Operatoren in der L_3 -Logik. Folgende Operatoren sind definiert:

Korrespondenz der Operatoren zwischen VDM-Logik und L_3 -Logik	
VDM-Logik	L_3 -Logik
$\wedge, \vee$	$\wedge, \vee$
$\neg$	$\neg$ starke Negation
$\rightarrow$	$\rightarrow$ starke Implikation
$\longleftrightarrow$	$\longleftrightarrow$ starke Äquivalenz

Tabelle 4.2: Operatoren in der VDM-SL ([Sch04a], S. 83)

Prädikate

In der VDM-Logik sind mehrere Prädikate spezifiziert. Im Folgenden soll jedoch zur Vereinfachung nur das $=$ -Prädikat spezifiziert werden. Die Prädikate entsprechen den logischen Vergleichsoperatoren in einer Programmiersprache ([Sch04a], S. 83).

$A =_3 B$	$B \in \mathbb{N}$	B hat den Wert $*\mathbb{N}$
$A \in \mathbb{N}$	$A = B$	$*\mathbb{B}$
A hat den Wert $*\mathbb{N}$	$*\mathbb{B}$	$*\mathbb{B}$

Tabelle 4.3: Prädikate in der VDM-SL ([Sch04a], S. 83)

Funktionen

Die typischen mathematischen arithmetischen Operatoren wie $+$, $-$, $*$, $/$ sind in der VDM-Logik als Funktionen spezifiziert. Im Folgenden sind die Funktionen für Division und Subtraktion auf den natürlichen Zahlen spezifiziert.

$$\left[\frac{n}{m}\right] = \begin{cases} \text{das kleinste } k \in \mathbb{N} \text{ mit } k \cdot m \geq n & \text{falls } m \neq 0 \\ *\mathbb{N} & \text{sonst} \end{cases}$$

$$[n - m] = \begin{cases} n - m & \text{falls } n \geq m \\ *\mathbb{N} & \text{sonst} \end{cases}$$

([Sch04a], S. 82)

4.1.2 Beispiel für die VDM-SL

Als Beispiel wird folgende VDM-logische Formel betrachtet:

$$\forall n, m ([\frac{n}{m}] = [m - n] \rightarrow 5n = 5m^2 - 5nm)$$

Wären jetzt $n, m \in \mathbb{N}$ und $m \geq n$, dann hätte die Formel den Wahrheitswert *true* bzw. *false*, je nachdem, welche Zahlen man für n oder m wählt. Tritt hingegen ein Fehler auf, da $m = 0$ oder $m \not\geq n$, wird das $=$ Prädikat nicht wahr auf der linken Seite der starken Implikation und bekommt den Wert $*\mathbb{B}$. Dadurch, dass die linke Seite der starken Implikation den Wert $*\mathbb{B}$ hat, bekommt die ganze Formel den Wert $*\mathbb{B}$, unabhängig davon, wie das rechte Prädikat $5n = 5m^2 - 5nm$ ausgewertet wird. ([Sch04a], S. 82)

Zusammenfassung und Ausblick

In dieser Seminararbeit wurden die grundlegenden Bestandteile einer m -wertigen Logik vorgestellt. Als konkretes Beispiel für eine m -wertige Logik wurde die dreiwertige Lukasiewicz-Logik präsentiert und diese anschließend um prädikatenlogische Aspekte erweitert. Zum Schluss wurde die Vienna Development Specification Language (VDM-SL) als Anwendungsbeispiel betrachtet und auf eine Beispielformel angewendet. Am Rande wurde auch das Tableauekalkül als Kalkül für endliche m -wertige Logiken vorgestellt.

Zum Abschluss der Arbeit sei erwähnt, dass lediglich die Grundlagen behandelt wurden. Viele weitere bedeutende Konzepte für m -wertige Logiken sind unberücksichtigt geblieben. Es wurde nur eine m -wertige Logik als Beispiel präsentiert, jedoch existieren zahlreiche weitere wichtige Kategorien von mehrwertigen Logiken, wie partielle Logiken, unendliche Logiken und unscharfe Logiken.

Es ist wichtig anzumerken, dass die Anwendungsbereiche für m -wertige Logiken vielfältig sind. Beispiele hierfür sind Spezifikationssprachen (zum Beispiel VDM-SL für die Spezifikation von Programmiersprachen), Testmustergenerierung (zum Beispiel für die Analyse von Schaltkreisen), Shape Analysis (zum Beispiel zur Verifikation von Programmen durch Graphen) und viele weitere praktische Anwendungen.

Literaturverzeichnis

- Sch04a. SCHMITT, P. H.: *Nichtklassische Logiken*, Juli 2004. Vorlesungsskript, Kapittel: Mehrwertige Logiken, Seiten: 19-124.
- Sch04b. SCHMITT, P. H.: *Nichtklassische Logiken*, Juli 2004. Vorlesungsskript, Kapitel: Wiederholung, Seiten: 8-18.

A

Selbstständigkeitserklärung

- ☐ Diese Arbeit habe ich selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet.
- ☐ Diese Arbeit wurde als Gruppenarbeit angefertigt. Meinen Anteil habe ich selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet.

Namen der Mitverfasser:

Meine eigene Leistung ist:

Datum

Unterschrift der Kandidatin/des Kandidaten