Vorkurs Theoretische Informatik

Einführung in die Grundideen, Mengenlehre und Aussagenlogik

Arbeitskreis Theo-Vorkurs

Montag, 7. Oktober 2024

Fachgruppe Informatik Universität Stuttgart

Aktuelle Folien:

Allgemeines

Wer sind wir?

- Fachgruppe Informatik
  - Unser Ziel:
    Das Leben von uns Studis während des Studiums angenehmer zu gestalten
  - organisieren Veranstaltungen (Grillen, Spieleabende, Vorkurse, ...)
  - verleihen Prüfungen aus den früheren Semestern
  - vertreten die studentische Sicht in offiziellen Gremien
  - ...und vieles mehr (es gibt z.B. einen 3D-Drucker)
- Arbeitskreis Theoretische Informatik
  - Teilmenge der Fachgruppe Informatik
  - haben diesen Vorkurs organisiert

Tipps zum Studium

- Nützliche Links:
  - Fachgruppe Informatik:
    `https://fius.de/`
  - Handouts und Foliensätze:
    `https://fius.de/index.php/studien-interessierte/vorkurs-theoretische-informatik/`
  - Materialien Ergänzung Theoretische Informatik 1 (Wintersemester 19/20):
    `https://fmi.uni-stuttgart.de/ti/teaching/w19/eti1/`
  - Ersti Telegram-Gruppe:
    `https://t.me/+78hAO9U16Ks0MTZi`
- E-Mail der Fachgruppe: fius@informatik.uni-stuttgart.de

Infos zum Online-Ablauf

Ablauf und Notfallplan

- Der Online-Vorkurs wird eine Übertragung aus/in einem Hörsaal sein
- Es wird zwischen Vorlesungs- und Aufgabephasen abgewechselt.
- Wir benutzen BigBlueButton - wenn ihr hier seid, wisst ihr das schon.
- Bei technischen Problemen, die sich nicht zügig beheben lassen, wechseln wir ggf. auf eine andere Plattform. Den Joinlink verschicken wir dann per Mail und stellen ihn auf `https://fius.de/index.php/studien-interessierte/vorkurs-theoretische-informatik/`.

Traut euch, Fragen zu stellen und mitzumachen.

Übersicht

Theoretische Informatik

Was ist eigentlich Theoretische Informatik?

- Theoretische Informatik ist die **formale** Herangehensweise an Probleme.
- Diese Probleme befassen sich unter Anderem mit den **formalen** Sprachen.

Anwendung der theoretischen Informatik

- Ist ein bestimmtes Problem lösbar, oder **können** wir gar keine Lösung finden?
- IT-Sicherheit / Kryptographie: Die Sicherheit bestimmter Algorithmen **beweisen**
- Reguläre Ausdrücke
- Künstliche Intelligenz
- Compilerbau
- ...und vieles mehr...

Theoretische Informatik in deinem Studium

Theoretische Informatik I ist Orientierungsprüfung für Informatik, Medieninformatik, Softwaretechnik und Data Science.

- Du musst diese Prüfung spätestens zum Ende des dritten Semester bestanden haben.
- Du musst spätestens zum Ende des zweiten Semesters eine der beiden Orientierungsprüfungen angetreten haben.
- Du kannst die schriftliche Prüfung einmal nachschreiben und hast dann noch einen mündlichen Versuch im selben Semester.

Kennt eure Prüfungsordnung!

Theoretische Informatik in deinem Studium

- Theoretische Informatik I
  Formale Sprachen und Automatentheorie (FSuA)
  - Dozent: Dr. Manfred Kufleitner
- Theoretische Informatik II
  Berechenbarkeit und Komplexität (BuK)
  - Dozent: Dr. Manfred Kufleitner
- Theoretische Informatik III
  Algorithmen (und Diskrete Strukturen) (AuDS)
  - Dozent: Prof. Stefan Funke

Altklausuren helfen bei der Prüfungsvorbereitung.

Fragt auch nach den Klausuren des alten Fachs.

Literatur der Vorlesung

Uwe Schöning: Theoretische Informatik - kurzgefasst
[€22,99]

- Die Vorlesung von Dr. Manfred Kufleitner richtet sich in weiten Teilen nach diesem Buch.

Boris Hollas: Grundkurs Theoretische Informatik: Mit Aufgaben und Anwendungen
[€27,99]

- Weniger formal, dafür intuitiver mit einigen Beispielen und Übungsaufgaben.

Dirk W. Hoffmann: Theoretische Informatik

- Wird auch gelegentlich empfohlen.

Literatur der Vorlesung

Uwe Schöning: Theoretische Informatik - kurzgefasst
[€0]

- Die Vorlesung von Dr. Manfred Kufleitner richtet sich in weiten Teilen nach diesem Buch.

Boris Hollas: Grundkurs Theoretische Informatik: Mit Aufgaben und Anwendungen
[€0]

- Weniger formal, dafür intuitiver mit einigen Beispielen und Übungsaufgaben.

Dirk W. Hoffmann: Theoretische Informatik

- Wird auch gelegentlich empfohlen.

Die Bücher sind alle in der Uni-Bib verfügbar, beim Schöning sollte man sich aber beeilen.

Wörter, Sprachen und Mengen

Mengen

- Was ist eine Menge?

Mengen

- Was ist eine Menge?

- Eine Menge
  - ist eine Sammlung von Zeugs
  - ist unsortiert
  - enthält keine Duplikate
  - wird mit geschweiften Klammern notiert

Beispiel

$\mathbb{N} = \{0, 1, 2, 3, \ldots\}$ = Menge der Natürlichen Zahlen

Studierende = {Julian, Joel, Fabian, Noah, . . . }

$\{1, 2\} = \{2, 1\} = \{1, 1, 2, 1, 1, 1\}$

$\varnothing = \{\}$ = leere Menge

Mengen

- Was ist eine Menge?
- Was ist ein Element?

Mengen

- Was ist eine Menge?
- Was ist ein Element?

- Ein Element ist ein Ding aus einer Menge.

Beispiel

1 ist ein Element der **Natürlichen Zahlen**

$\mathbf{1} \in \mathbb{N}$

Julian ist ein Element aus der Menge der **Studierenden**

Julian $\in$ **Studierende**

a ist in der Menge $\{\mathbf{u}, \mathbf{v}, \mathbf{w}\}$ nicht enthalten

$\mathbf{a} \notin \{\mathbf{u}, \mathbf{v}, \mathbf{w}\}$

Mengen

- Was ist eine Menge?
- Was ist ein Element?
- Was ist eine Teilmenge?

Mengen

- Was ist eine Menge?
- Was ist ein Element?
- Was ist eine Teilmenge?

- Eine Teilmenge ist eine spezielle Auswahl von Elementen einer Menge.

Beispiel

$\{1, 2, 3\}$ ist eine Teilmenge der Natürlichen Zahlen

$\{1, 2, 3\} \subseteq \mathbb{N}$

{**Julian**} ist eine Teilmenge der **Studierenden**

{**Julian**} $\subseteq$ **Studierende**

Mengen - Mal anders

Ein paar Definitionen

Eine nichtleere Menge einstelliger Symbole nennen wir Alphabet. Es wird oft dargestellt durch den Bezeichner Σ.

Beispiele

- $\Sigma = \{a, b\}$
- $\Sigma = \{0, 1\}$
- $\Sigma = \{\text{Rechts, Links, Vorwärts, Rückwärts, Start, Stopp, Pause}\}$

Mengen - Mal anders

Ein paar Definitionen

Auf einem Alphabet können wir die Operation $\cdot$, genannt Konkatenation, ausüben.

$\longrightarrow$ zum Beispiel ist dann $a \cdot b = ab$

Eine beliebig lange Kette an Symbolen aus dem Alphabet nennen wir ein Wort.

Beispiele

- *abba* ist ein *Wort* über dem *Alphabet* $\Sigma = \{a, b\}$
- 10011101 ist ein *Wort* über dem *Alphabet* $\Sigma = \{0, 1\}$
- StartVorwärtsRechtsVorwärtsStopp ist ein Wort über $\Sigma = \{\text{Rechts, Links, Vorwärts, Rückwärts, Start, Stopp, Pause}\}$

Wortlängen

Wortlänge und das leere Wort

Eine endlich lange Kette an Symbolen aus dem Alphabet nennen wir ein Wort.

- Wort der Länge 3: z.B. aaa, aba, 110, RechtsPauseStopp . . .
- Wort der Länge 2: z.B. aa, ab, 00, StartVorwärts . . .
- Wort der Länge 1: z.B. a, b, 1, Links . . .
- Wort der Länge 0: ε

Wir schreiben $|w|$ um Länge des Wortes w abzukürzen.

Wortlängen

Wortlänge und das leere Wort

Eine endlich lange Kette an Symbolen aus dem Alphabet nennen wir ein Wort.

- Wort der Länge 3: z.B. aaa, aba, 110, RechtsPauseStopp ...
- Wort der Länge 2: z.B. aa, ab, 00, StartVorwärts ...
- Wort der Länge 1: z.B. a, b, 1, Links ...
- Wort der Länge 0: ε

Wir schreiben $|w|$ um Länge des Wortes w abzukürzen.

Um nur ein Symbol (z.B. a) zu zählen verwenden wir $|w|_a$.

Wortlängen

Wortlänge und das leere Wort

Eine endlich lange Kette an Symbolen aus dem Alphabet nennen wir ein Wort.

- Wort der Länge 3: z.B. aaa, aba, 110, RechtsPauseStopp ...
- Wort der Länge 2: z.B. aa, ab, 00, StartVorwärts ...
- Wort der Länge 1: z.B. a, b, 1, Links ...
- Wort der Länge 0: ε

ε *(„Epsilon“)* nennen wir das „leere Wort“.

- Vergleich: Es ist vergleichbar mit einem leerem String,
  also: $'''' = \varepsilon$
- Achtung: Das leere Wort kann kein Teil eines Alphabets sein,
  da es nicht einstellig ist. Es hat die Wortlänge $|\varepsilon| = 0$

Neutrales Element der Konkatenation

Achtung

Wir können bei der Konkatenation auch das leere Wort anhängen. Es verhält sich hierbei als das neutrale Element.

d.h. für ein beliebiges Wort w, ist $w \cdot \varepsilon = \varepsilon \cdot w = w$

Vergleich

- Bei der Addition von Zahlen ist die 0 das neutrale Element
  $n + 0 = 0 + n = n$
- Bei der Multiplikation von Zahlen ist die 1 das neutrale Element
  $x * 1 = 1 * x = x$

Was heißt das?

Bei der Konkatenation $a \cdot a \cdot \varepsilon \cdot a$ entsteht das Wort aaa mit $|aaa| = 3$.

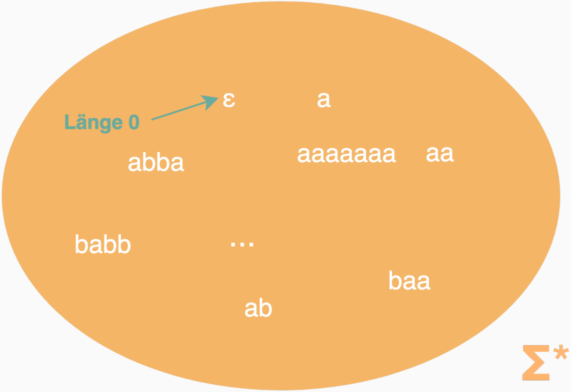

Abbildung 1: Menge von allen Kombinationen der Elemente von Σ heißt Σ^*

Das heißt...

Sei $\Sigma = \{a, b\}$ unser Alphabet.

Wir beschreiben die Menge, die alle Möglichkeiten enthält Elemente aus Σ zu *konkatenieren* mit $\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aba, \ldots\}$.

Das heißt...

Sei $\Sigma = \{a, b\}$ unser Alphabet.

Wir beschreiben die Menge, die alle Möglichkeiten enthält Elemente aus Σ zu *konkatenieren* mit $\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aba, \dots\}$.

Achtung

M^* über einer beliebigen Menge M enthält immer das leere Wort ε!

Sogar wenn $M = \{\} = \emptyset$.

Denkpause

Aufgaben

Nenne jeweils 5 der kürzesten Elemente aus Σ^* für die folgenden Alphabete Σ:

Normal

- $\Sigma = \{a\}$
- $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

Etwas schwerer

- Σ = {0, Mei, Totoro}
- Σ = {☺, ☹}

Lösungen

Mögliche Lösungen sind …

- $\varepsilon, a, aa, aaa, aaaa \in \{a\}^*$

Lösungen

Mögliche Lösungen sind ...

- $\varepsilon, a, aa, aaa, aaaa \in \{a\}^*$
- $\varepsilon, 0, 1, 2, 3 \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}^*$

Lösungen

Mögliche Lösungen sind …

- $\varepsilon, a, aa, aaa, aaaa \in \{a\}^*$
- $\varepsilon, 0, 1, 2, 3 \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}^*$
- $\varepsilon, 0, Mei, Totoro, MeiTotoro \in \{0, Mei, Totoro\}^*$

Lösungen

Mögliche Lösungen sind …

- $\varepsilon, a, aa, aaa, aaaa \in \{a\}^*$
- $\varepsilon, 0, 1, 2, 3 \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}^*$
- $\varepsilon, 0, \mathit{Mei}, \mathit{Totoro}, \mathit{MeiTotoro} \in \{0, \mathit{Mei}, \mathit{Totoro}\}^*$
- $\varepsilon, \smiley, \frownie, \smiley\smiley, \smiley\frownie \in \{\smiley, \frownie\}^*$

Verständnisabfrage

Denke kurz über folgende Aufgabe nach...

Schwer

Welche Wörter sind in M^* enthalten, wenn $M = \emptyset$ gilt?

Lösungen

- In $M^* = \{\ \}^*$ ist **nur** das leere Wort ε enthalten.

Sprachen

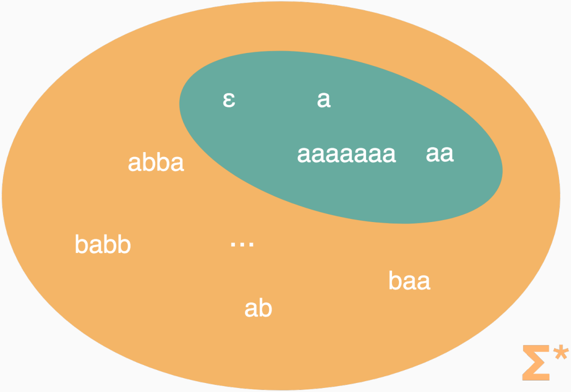

Abbildung 2: Teilmengen unserer Obermenge (hier Σ^*) nennen wir Sprachen

Sprachen beschreiben

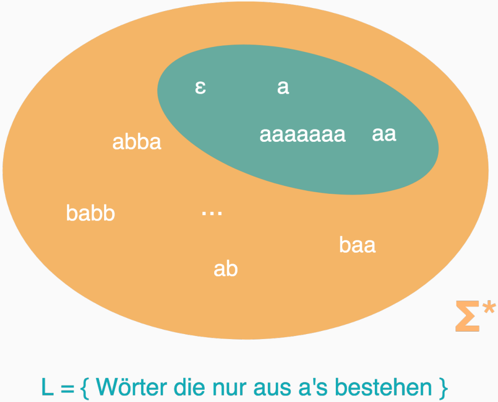

Abbildung 3: Manche Sprachen können wir mit Regeln beschreiben

Sprachen beschreiben

Beispiele für Sprachen in Mengenschreibweise

- $L_1 = \{a^n \mid n \in \mathbb{N}\}$
- $L_2 = \{a^n \mid n \equiv 0 \pmod 2, n \in \mathbb{N}\}$
- $L_3 = \{uv \mid u \in \{a, b\}^*,\ v \in \{a\}\}$
- $L_4 = \{w \mid |w|_a = 3\}$

Was soll das alles? Mehr dazu nach der Pause :)

Murmelpause

Mengenschreibweise

Wie sprechen wir das?

$L_2 = \{a^n \mid n \in \mathbb{N}\}$

Die Sprache L_2 enthält alle Wörter a^n, für die gilt: n stammt aus der Menge der natürlichen Zahlen.

Achtung

In der theoretischen Informatik enthält $\mathbb{N}$ ($\mathbb{N}$ ist die Menge der natürlichen Zahlen) die Zahl 0.

Wie schreiben wir das?

- Viele Zeichen hintereinander (konkateniert) können auch einfacher geschrieben werden.

$$
\begin{aligned}
a^0 &= \varepsilon \\
a^1 &= a \\
a^2 &= a \cdot a = aa \\
a^3 &= a \cdot a \cdot a = aaa \\
&\vdots \\
a^n &= \underbrace{a \cdot a \cdot \ldots \cdot a}_{\text{n-mal}}
\end{aligned}
$$

Wie schreiben wir das?

- $L_1 = \{0, 2, 4, 6, 8, \dots\} = \{x \mid \text{x ist gerade}\}$
  $= \{x \mid \text{Es gibt eine Zahl } k \in \mathbb{N} : 2k = x\}$

Wie schreiben wir das?

- $L_1 = \{0, 2, 4, 6, 8, \dots\} = \{x \mid \text{x ist gerade}\}$
  $= \{x \mid \text{Es gibt eine Zahl } k \in \mathbb{N} : 2k = x\}$
- $L_2 = \{a^n \mid n \in \mathbb{N}\} = \{\varepsilon, a, aa, aaa, aaaa, \dots\}$

Wie schreiben wir das?

- $L_1 = \{0, 2, 4, 6, 8, \dots\} = \{x \mid \text{x ist gerade}\}$
  $= \{x \mid \text{Es gibt eine Zahl } k \in \mathbb{N} : 2k = x\}$
- $L_2 = \{a^n \mid n \in \mathbb{N}\} = \{\varepsilon, a, aa, aaa, aaaa, \dots\}$
- $L_3 = \{a^n b^n \mid n \in \mathbb{N}\} = \{\varepsilon, ab, aabb, aaabbb, aaaabbbb, \dots\}$

Wie schreiben wir das?

- $L_1 = \{0, 2, 4, 6, 8, \dots\} = \{x \mid \text{x ist gerade}\}$
  $= \{x \mid \text{Es gibt eine Zahl } k \in \mathbb{N} : 2k = x\}$
- $L_2 = \{a^n \mid n \in \mathbb{N}\} = \{\varepsilon, a, aa, aaa, aaaa, \dots\}$
- $L_3 = \{a^n b^n \mid n \in \mathbb{N}\} = \{\varepsilon, ab, aabb, aaabbb, aaaabbbb, \dots\}$
- $L_4 = \{a^n w \mid n \in \mathbb{N}, w = bccb\} = \{bccb, abccb, aabccb, \dots\}$
  L_4 endet nach einer beliebigen Anzahl von a's immer mit bccb

Wie schreiben wir das?

- $L_1 = \{0, 2, 4, 6, 8, \dots\} = \{x \mid \text{x ist gerade}\}$
  $= \{x \mid \text{Es gibt eine Zahl } k \in \mathbb{N} : 2k = x\}$
- $L_2 = \{a^n \mid n \in \mathbb{N}\} = \{\varepsilon, a, aa, aaa, aaaa, \dots\}$
- $L_3 = \{a^n b^n \mid n \in \mathbb{N}\} = \{\varepsilon, ab, aabb, aaabbb, aaaabbbb, \dots\}$
- $L_4 = \{a^n w \mid n \in \mathbb{N}, w = bccb\} = \{bccb, abccb, aabccb, \dots\}$
  L_4 endet nach einer beliebigen Anzahl von a's immer mit bccb
- $L_5 = \{w \mid |w| = 2, w \in \{a, b\}^*\} = \{aa, bb, ab, ba\}$
  Wörter der Länge 2 aus $\{a, b\}^*$

Wie schreiben wir das?

- $L_1 = \{0, 2, 4, 6, 8, \dots\} = \{x \mid \text{x ist gerade}\}$
  $= \{x \mid \text{Es gibt eine Zahl } k \in \mathbb{N} : 2k = x\}$
- $L_2 = \{a^n \mid n \in \mathbb{N}\} = \{\varepsilon, a, aa, aaa, aaaa, \dots\}$
- $L_3 = \{a^n b^n \mid n \in \mathbb{N}\} = \{\varepsilon, ab, aabb, aaabbb, aaaabbbb, \dots\}$
- $L_4 = \{a^n w \mid n \in \mathbb{N}, w = bccb\} = \{bccb, abccb, aabccb, \dots\}$
  L_4 endet nach einer beliebigen Anzahl von a's immer mit bccb
- $L_5 = \{w \mid |w| = 2, w \in \{a, b\}^*\} = \{aa, bb, ab, ba\}$
  Wörter der Länge 2 aus $\{a, b\}^*$
- $L_6 = \{w \mid |w|_a = 2, w \in \{a, b\}^*\}$
  Wörter mit genau 2 a's und beliebig vielen b's aus $\{a, b\}^*$

Aufgaben

Findet Wörter aus den folgenden Sprachen

Normal

- $L_1 = \{a\}$
- $L_2 = \{uv \mid u \in \{a, b\}^*,\ v \in \{c, d\}\}$
- $L_3 = \{w \mid |w| = 3, w \in \{a, b, c\}^*\}$

Etwas Schwerer

- $L_4 = \{a^n \mid n \equiv 1 \pmod 3, n \in \mathbb{N}\}$
- $L_5 = \{w \mid |w|_a = 3, |w|_b = 1, w \in \{a, b, c\}^*\}$
- $L_6 = \{uv \mid u \in \{◀, ▲, ▶, ▼\}^*,\ v \in \{\text{STOP}\}\}$
- $L_7 = \{w \mid |w| = 2, w \in \{a, b\}\}$

Anmerkung: $x \equiv y \pmod n \iff n$ teilt $(x - y)$ ohne Rest $\iff x = m \cdot n + y$ mit $x, y, n, m \in \mathbb{Z}$

Lösungen

- L_1: Enthält **nur** das einzelne Wort a!

Lösungen

- L_1: Enthält **nur** das einzelne Wort a!
- L_2: z.B. $c, d, ac, bc, aaac, abababad, \dots$
  Wort besteht aus zwei Teilen: u z.B. ε, a, b, $ababa$, ...
  v ist entweder c oder d!

Lösungen

- L_1: Enthält **nur** das einzelne Wort a!
- L_2: z.B. $c, d, ac, bc, aaac, abababad, \ldots$
  Wort besteht aus zwei Teilen: u z.B. ε, a, b, $ababa$, ...
  v ist entweder c oder d!
- L_3: enthält alle Wörter der Länge 3, deren Buchstaben nur a, b oder c sind.
  $\longrightarrow$ aaa, aab, aba, abb, abc, acb, acc, baa, bab, cbb, ...

Lösungen

- L_1: Enthält **nur** das einzelne Wort a!
- L_2: z.B. $c, d, ac, bc, aaac, abababad, \dots$
  Wort besteht aus zwei Teilen: u z.B. ε, a, b, $ababa$, ...
  v ist entweder c oder d!
- L_3: enthält alle Wörter der Länge 3, deren Buchstaben nur a, b oder c sind.
  $\longrightarrow$ aaa, aab, aba, abb, abc, acb, acc, baa, bab, cbb, ...
- $L_4 = \{a, aaaa, aaaaaaa, \dots\}$
  Wörter deren Länge durch 3 geteilt den Rest 1 ergeben.

Lösungen

- L_1: Enthält **nur** das einzelne Wort a!
- L_2: z.B. $c, d, ac, bc, aaac, abababad, \ldots$
  Wort besteht aus zwei Teilen: u z.B. ε, a, b, $ababa$, ...
  v ist entweder c oder d!
- L_3: enthält alle Wörter der Länge 3, deren Buchstaben nur a, b oder c sind.
  $\longrightarrow$ aaa, aab, aba, abb, abc, acb, acc, baa, bab, cbb, ...
- $L_4 = \{a, aaaa, aaaaaaa, \ldots\}$
  Wörter deren Länge durch 3 geteilt den Rest 1 ergeben.
- $L_5 = \{caaba, cccbaaa, abaca, aaab, \ldots\}$
  genau 3 a's, genau 1 b, beliebig viele c's, keine Sortierung

Lösungen

- L_1: Enthält **nur** das einzelne Wort a!
- L_2: z.B. $c, d, ac, bc, aaac, abababad, \dots$
  Wort besteht aus zwei Teilen: u z.B. ε, a, b, $ababa$, ...
  v ist entweder c oder d!
- L_3: enthält alle Wörter der Länge 3, deren Buchstaben nur a, b oder c sind.
  $\longrightarrow$ aaa, aab, aba, abb, abc, acb, acc, baa, bab, cbb, ...
- $L_4 = \{a, aaaa, aaaaaaa, \dots\}$
  Wörter deren Länge durch 3 geteilt den Rest 1 ergeben.
- $L_5 = \{caaba, cccbaaa, abaca, aaab, \dots\}$
  genau 3 a's, genau 1 b, beliebig viele c's, keine Sortierung
- L_6 = {STOP, ◀STOP, ▲STOP,... , ▼◀▼STOP,...}

Lösungen

- L_1: Enthält **nur** das einzelne Wort a!
- L_2: z.B. $c, d, ac, bc, aaac, abababad, \ldots$
  Wort besteht aus zwei Teilen: u z.B. ε, a, b, $ababa$, ...
  v ist entweder c oder d!
- L_3: enthält alle Wörter der Länge 3, deren Buchstaben nur a, b oder c sind.
  $\longrightarrow$ $aaa, aab, aba, abb, abc, acb, acc, baa, bab, cbb$, ...
- $L_4 = \{a, aaaa, aaaaaaa, \ldots\}$
  Wörter deren Länge durch 3 geteilt den Rest 1 ergeben.
- $L_5 = \{caaba, cccbaaa, abaca, aaab, \ldots\}$
  genau 3 a's, genau 1 b, beliebig viele c's, keine Sortierung
- L_6 = {🛑, ◀🛑, ▲🛑,... , ▼◀▼🛑,...}
- $L_7 = \emptyset$
  L_7 enthält **gar kein** Wort!

Mengenoperationen

Schnitt - $A \cap B$

Gegeben zwei Mengen A und B.

In der Schnittmenge liegt alles, das in Menge A **und** in Menge B ist.

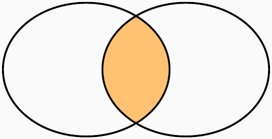

Abbildung 4: Veranschaulichung der Schnittmenge

Vereinigung - $A \cup B$

Gegeben zwei Mengen A und B.
In der Vereinigung liegt alles, das nur in A, nur in B **oder** in beiden Mengen liegt.

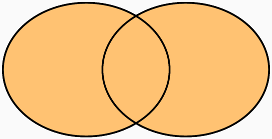

Abbildung 5: Veranschaulichung der Vereinigung

Komplement - $\overline{A}$

Gegeben sei eine Menge A.
Im Komplement der Menge A liegen alle Elemente, die in der Obermenge (z.B. Σ^*), aber nicht in der Menge A selbst liegen.

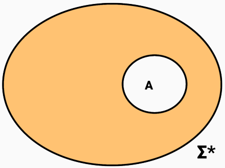

Abbildung 6: Veranschaulichung des Komplements

Mengenoperationen - Komplement

Komplement - $\overline{A}$

Gegeben sei eine Menge A.
Im Komplement der Menge A liegen alle Elemente, die in der Obermenge (z.B. Σ^*), aber nicht in der Menge A selbst liegen.

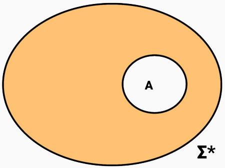

Abbildung 6: Veranschaulichung des Komplements

Anmerkung: Kann auch geschrieben werden als $\Sigma^* \setminus A$.
(gesprochen Σ^* *„ohne“* A)

Berechne folgende Mengen

Normal

- $M_1 = \{a\} \cup \{b\}$
- $M_2 = \{\} \cap \{u, v, w\}$
- $M_3 = \mathbb{N} \cup \mathbb{Z}$
- $M_4 = \overline{\{a^n \mid n \text{ ist gerade}\}}$, über dem Alphabet $\{a\}$

Schwer bis sehr schwer

- $M_5 = \{a, b, c\} \cap \{a, \{b, c\}\}$
- $M_6 = \{u \mid |u| \equiv 0 \pmod 2, u \in \{a, b\}^*\}$
  $\cup \{v \mid |v| \equiv 0 \pmod 4, v \in \{a, b\}^*\}$
- $M_7 = \overline{\{a^n \mid n \text{ ist gerade}\}}$, über dem Alphabet $\{a, b\}$

Lösungen

- $M_1 = \{a, b\}$

Lösungen

- $M_1 = \{a, b\}$
- $M_2 = \emptyset$

Lösungen

- $M_1 = \{a, b\}$
- $M_2 = \emptyset$
- $M_3 = \mathbb{Z}$

Lösungen

- $M_1 = \{a, b\}$
- $M_2 = \emptyset$
- $M_3 = \mathbb{Z}$
- $M_4 = \{a^n \mid \text{n ist ungerade}\}$

Lösungen

- $M_1 = \{a, b\}$
- $M_2 = \emptyset$
- $M_3 = \mathbb{Z}$
- $M_4 = \{a^n \mid \text{n ist ungerade}\}$
- $M_5 = \{a\}$

Lösungen

- $M_1 = \{a, b\}$
- $M_2 = \emptyset$
- $M_3 = \mathbb{Z}$
- $M_4 = \{a^n \mid \text{n ist ungerade}\}$
- $M_5 = \{a\}$
- $M_6 = \{u \mid |u| \equiv 0 \pmod 2, u \in \{a, b\}^*\}$

Lösungen

- $M_1 = \{a, b\}$
- $M_2 = \emptyset$
- $M_3 = \mathbb{Z}$
- $M_4 = \{a^n \mid \text{n ist ungerade}\}$
- $M_5 = \{a\}$
- $M_6 = \{u \mid |u| \equiv 0 \pmod 2, u \in \{a, b\}^*\}$
- $M_7 = \{a^n \mid n \text{ ist ungerade}\} \cup \{w \mid w \in \{a, b\}^*, |w|_b \geq 1\}$

Aussagenlogik

Was ist Aussagenlogik?

Aussagen

- Paris ist die Hauptstadt von Frankreich
- Totoro ist ein Mensch
- $5 \in \mathbb{N}$
- $5 = 8$
- $u \in \{u, v, w\}$

Eine Aussage A ist entweder **wahr** oder **falsch**.

Was ist Aussagenlogik?

Das sind keine Aussagen

- Macht theoretische Informatik Spaß?
- Geh dein Zimmer aufräumen!
- Wie oft trifft Mei auf Totoro?
- $(x + y)^2 + 1$
- $\{a, b, c\}$
- ...

Diesen Sätzen können wir keinen eindeutigen Wahrheitswert **wahr** oder **falsch** zuordnen.

Was ist Aussagenlogik?

Wozu brauchen wir das?

- Wir untersuchen, wie wir Aussagen verknüpfen können.
- Damit ziehen wir formale Schlüsse und führen Beweise.

Logische Operationen

Wir können Aussagen verändern oder durch Operationen zu neuen Aussagen verbinden.

- A: Totoro möchte Nüsse.
- B: Totoro will schlafen.

Grundoperationen

- **Und**: $A \wedge B \rightsquigarrow$ Totoro möchte Nüsse und will schlafen.
  Analog: M: $M_1 \cap M_2$, Jedes Element aus M liegt in M_1 **und** in M_2

Logische Operationen

Wir können Aussagen verändern oder durch Operationen zu neuen Aussagen verbinden.

- A: Totoro möchte Nüsse.
- B: Totoro will schlafen.

Grundoperationen

- **Und**: $A \wedge B \rightsquigarrow$ Totoro möchte Nüsse und will schlafen.
- **Oder**: $A \vee B \rightsquigarrow$ Totoro möchte Nüsse oder will schlafen.
  Anmerkung: Inklusives „oder“, kein „entweder oder“
  Das heißt, es können auch beide Aussagen wahr sein.
  Analog: M: $M_1 \cup M_2$, Jedes Element aus M liegt in M_1 **oder** in M_2

Logische Operationen

Wir können Aussagen verändern oder durch Operationen zu neuen Aussagen verbinden.

- A: Totoro möchte Nüsse.
- B: Totoro will schlafen.

Grundoperationen

- **Und**: $A \land B \rightsquigarrow$ Totoro möchte Nüsse und will schlafen.
- **Oder**: $A \lor B \rightsquigarrow$ Totoro möchte Nüsse oder will schlafen.
- **Nicht**: $\neg A \rightsquigarrow$ Totoro möchte keine Nüsse.
  Analog: M: $\overline{M_1}$, Jedes Element aus M liegt **nicht** in M_1

Überblick: Mengenoperationen

Auf Mengen A, B lassen sich verschiedene Mengenoperationen ausführen.

Mengenoperationen

- Schnitt: $A \cap B$
- Vereinigung: $A \cup B$
- Komplement: $\overline{A}$

Überblick: Logische Operationen

Auf Aussagen A, B lassen sich verschiedene logische Operationen ausführen.

Logische Operationen

- Logisches Und: $A \land B$
- Logisches Oder: $A \lor B$
- Logisches Nicht: $\neg A$

Logische Operationen vs. Mengenoperationen

Mengenoperationen und logische Operationen dürfen nicht verwechselt werden.

A: $5 \in \mathbb{N}$ $\quad$ B: Es regnet

C: $\{w \mid |w| = 2\}$ $\quad$ D: {a,b,c,x,y}

Welche dieser Verknüpfungen sind zulässig?

1. $A \wedge B$
2. $A \vee C$
3. $C \cap D$
4. $A \wedge B \cup C$

Logische Operationen vs. Mengenoperationen

1. Zulässig

Logische Operationen vs. Mengenoperationen

1. Zulässig
2. Nicht zulässig

Logische Operationen vs. Mengenoperationen

1. Zulässig
2. Nicht zulässig
3. Zulässig

Logische Operationen vs. Mengenoperationen

1. Zulässig
2. Nicht zulässig
3. Zulässig
4. Nicht zulässig

Logische Operationen: Implikation

$A \implies B$

- „Wenn A wahr ist, dann muss auch B wahr sein."
- kurz: „**Wenn** A, **dann** B."
- Wenn A falsch ist können wir keine Aussage über B treffen.
- $A \implies B$ ist dieselbe Aussage wie $\neg A \vee B$

$A \implies B$

- „Wenn A wahr ist, dann muss auch B wahr sein."
- kurz: „**Wenn** A**, dann** B."
- Wenn A falsch ist können wir keine Aussage über B treffen.
- $A \implies B$ ist dieselbe Aussage wie $\neg A \lor B$

Kontraposition

- $\neg B \implies \neg A$ ist die Kontraposition zu $A \implies B$
- "Wenn B falsch ist, dann muss A auch falsch (gewesen) sein"
- Die Implikation und ihre Kontraposition haben den selben Wahrheitswert

Denkpause

Aufgaben

Die folgenden Teilaufgaben bestehen aus einer Aussage, einem Geschehen und aus einer Folgerung. Welche der Folgerungen sind richtig, unter der Annahme, dass die Aussagen wahr sind?

Normal bis Schwer

- Aussage: "Wenn Lukas nicht im Vorkurs, schläft er noch"
  Geschehen: Lukas ist nicht im Vorkurs.
  Folgerung: Also schläft er noch.
- Aussage: "Wenn Mei nicht rennt, bekommt sie den Bus nicht"
  Geschehen: Mei bekommt den Bus.
  Folgerung: Also ist sie gerannt.
- Aussage: "Wenn Tobi auf die Prüfung nicht lernt, besteht er nicht"
  Geschehen: Tobi hat auf die Prüfung gelernt.
  Folgerung: Also besteht er.

Lösungen

Normal bis Schwer

- Aussage: "Wenn Lukas nicht im Vorkurs, schläft er noch"
  Geschehen: Lukas ist nicht im Vorkurs.
  Folgerung: Also schläft er noch.

Die Folgerung ist richtig. Laut Aussage ist Lukas im Vorkurs oder schläft noch (oder beides). Wenn er also nicht im Vorkurs ist, bleibt nur noch die Option übrig, dass er noch schläft.

Lösungen

Normal bis Schwer

- Aussage: "Wenn Mei nicht rennt, bekommt sie den Bus nicht"
  Geschehen: Mei bekommt den Bus.
  Folgerung: Also ist sie gerannt.

Die Folgerung ist richtig. Die zweite Aussage ist die Kontraposition der ersten Aussage. Die Kontraposition einer Aussage ist wahr gdw. die Aussage selbst wahr ist.

Lösungen

Normal bis Schwer

- Aussage: "Wenn Tobi auf die Prüfung nicht lernt, besteht er nicht"
  Geschehen: Tobi hat auf die Prüfung gelernt.
  Folgerung: Also besteht er.

Die Folgerung ist falsch. Es ist auch möglich, dass Tobi lernt und nicht besteht!

Logische Operationen: Äquivalenz

$A \iff B$

- „A ist wahr, **genau dann wenn** B wahr ist."
- kurz: „A gdw. B"
- A und B müssen den selben Wahrheitswert haben.
- $A \iff B$ ist dieselbe Aussage wie $(A \implies B) \land (B \implies A)$

Aufgaben

Berechne den Wahrheitswert folgender Aussagen.

Normal

- A_1: $5 \in \mathbb{N} \land a \in \{a, b, c\}$
- A_2: $0 \in \mathbb{N} \lor a \in \{a, b, c, d\}$
- A_3: $A_1 \iff A_2$

Etwas Schwerer

- A_4: $(\emptyset = \emptyset^*) \implies (a \in \emptyset)$
- A_5: $(a \notin \emptyset) \implies (\emptyset = \emptyset^*)$
- A_6: $A_4 \iff A_5$
- A_7: $(7 \in \{1, 2, 7, 9\}) \cap (2 = 7 - 5)$
- A_8: Wenn es Totoro in echt gibt,
  dann frisst Totoro kleine Kinder.

Lösungen

- A_1: wahr

Lösungen

- A_1: wahr
- A_2: wahr

Lösungen

- A_1: wahr
- A_2: wahr
- A_3: wahr

Lösungen

- A_1: wahr
- A_2: wahr
- A_3: wahr
- A_4: wahr

Lösungen

- A_1: wahr
- A_2: wahr
- A_3: wahr
- A_4: wahr
- A_5: falsch

Lösungen

- A_1: wahr
- A_2: wahr
- A_3: wahr
- A_4: wahr
- A_5: falsch
- A_6: falsch

Lösungen

- A_1: wahr
- A_2: wahr
- A_3: wahr
- A_4: wahr
- A_5: falsch
- A_6: falsch
- A_7: Das ist keine Aussage, da der Schnitt ($\cap$) verwendet wurde um zwei Aussagen miteinander zu verknüpfen

Lösungen

- A_1: wahr
- A_2: wahr
- A_3: wahr
- A_4: wahr
- A_5: falsch
- A_6: falsch
- A_7: Das ist keine Aussage, da der Schnitt ($\cap$) verwendet wurde um zwei Aussagen miteinander zu verknüpfen
- A_8: wahr

Wir haben zwei Aussagen A und B. Wir nehmen nun an, A sei wahr. Wenn wir zeigen, dass dann auch B wahr ist, wissen wir, dass $A \implies B$ gilt.

Beispiel

1. Wir wollen zeigen, dass für eine ganze Zahl x die Implikation $3 = x - 2 \implies x = 5$ gilt.

Anwendung der Implikation

Wir haben zwei Aussagen A und B. Wir nehmen nun an, A sei wahr. Wenn wir zeigen, dass dann auch B wahr ist, wissen wir, dass $A \implies B$ gilt.

Beispiel

1. Wir wollen zeigen, dass für eine ganze Zahl x
   die Implikation $3 = x - 2 \implies x = 5$ gilt.
2. Wir nehmen an, dass die linke Aussage wahr ist...

Anwendung der Implikation

Wir haben zwei Aussagen A und B. Wir nehmen nun an, A sei wahr. Wenn wir zeigen, dass dann auch B wahr ist, wissen wir, dass $A \implies B$ gilt.

Beispiel

1. Wir wollen zeigen, dass für eine ganze Zahl x die Implikation $3 = x - 2 \implies x = 5$ gilt.
2. Wir nehmen an, dass die linke Aussage wahr ist…
3. …und zeigen, dass dann die rechte Aussage gilt.

Anwendung der Implikation

Wir haben zwei Aussagen A und B. Wir nehmen nun an, A sei wahr. Wenn wir zeigen, dass dann auch B wahr ist, wissen wir, dass $A \implies B$ gilt.

Beispiel

1. Wir wollen zeigen, dass für eine ganze Zahl x die Implikation $3 = x - 2 \implies x = 5$ gilt.
2. Wir nehmen an, dass die linke Aussage wahr ist...
3. ...und zeigen, dass dann die rechte Aussage gilt.
4. $(3 = x - 2) \implies (3 + 2 = x) \implies (5 = x) \implies (x = 5)$

Anwendung der Implikation

Wir haben zwei Aussagen A und B. Wir nehmen nun an, A sei wahr. Wenn wir zeigen, dass dann auch B wahr ist, wissen wir, dass $A \implies B$ gilt.

Beispiel

1. Wir wollen zeigen, dass für eine ganze Zahl x die Implikation $3 = x - 2 \implies x = 5$ gilt.
2. Wir nehmen an, dass die linke Aussage wahr ist...
3. ...und zeigen, dass dann die rechte Aussage gilt.
4. $(3 = x - 2) \implies (3 + 2 = x) \implies (5 = x) \implies (x = 5)$
5. Also folgt die rechte Aussage aus der Linken.

Anwendung der Implikation

Wir haben zwei Aussagen A und B. Wir nehmen nun an, A sei wahr. Wenn wir zeigen, dass dann auch B wahr ist, wissen wir, dass $A \implies B$ gilt.

Beispiel

1. Wir wollen zeigen, dass für eine ganze Zahl x die Implikation $3 = x - 2 \implies x = 5$ gilt.
2. Wir nehmen an, dass die linke Aussage wahr ist…
3. …und zeigen, dass dann die rechte Aussage gilt.
4. $(3 = x - 2) \implies (3 + 2 = x) \implies (5 = x) \implies (x = 5)$
5. Also folgt die rechte Aussage aus der Linken.
6. Somit gilt die Implikation. $\square$

Wiederholung

Das können wir jetzt beantworten

Einführung

- Theoretische Informatik ist ganz schön wichtig...
- ...für mein Studium.

Das können wir jetzt beantworten

Mengen, Sprachen, Elemente

- Was ist eine Menge?
- Was ist eine Sprache?
- Was sind Elemente einer Sprache/Menge?

Das können wir jetzt beantworten

Alphabete, Σ^*

- Was ist ein Alphabet?, Was ist ein Wort?
- Wie funktioniert die Konkatenation?
- Was ist der Unterschied zwischen Σ und Σ^*?
- Das leere Wort: Welches ist das *kleinste* Alphabet mit $\varepsilon \in \Sigma^*$?
- Bilden von Σ^* für gegebenes Alphabet Σ

Das können wir jetzt beantworten

Operationen auf Mengen

- Wie funktionieren Vereinigung, Schnitt und Komplement?
- Wie bilde ich die Vereinigung oder Schnittmenge zweier Mengen?
- Wie bilde ich das Komplement einer Menge?
- Wie kann ich Sprachen formal beschreiben?
- Hantieren mit verschiedenen seltsamen Mengen und den Verknüpfungen

Das können wir jetzt beantworten

Logik

- Was ist eine Aussage?
- Was bedeuten $\wedge$, $\vee$ und $\neg$?
- Wie funktioniert die Implikation?
- Wie funktioniert die Äquivalenz?

Noch Fragen?

Glossar

Abk.	Bedeutung	Was?!
$\mathbb{N}$	natürliche Zahlen (mit 0)	In der theoretischen Informatik enthält $\mathbb{N}$ die 0: $\mathbb{N} = \{0, 1, 2, 3, \ldots\}$ (Auch $\mathbb{N}_0$)
$\mathbb{Z}$	ganze Zahlen	$\mathbb{Z} = \{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\}$
Σ	Sigma	mit diesem Zeichen wird oft das Alphabet (die Menge an verwendbaren Symbolen) repräsentiert
Σ^*	Sigma Stern	Menge aller Möglichkeiten Elemente aus Σ hintereinander zu schreiben
ε	leeres Wort	Wort (über bel. Alphabet) mit der Länge 0 ($\lvert\varepsilon\rvert = 0$), in allen Σ^* enthalten.
$\emptyset$	{}	leere Menge
$a \mid b$	teilt	a ist Teiler von b, d.h. a teilt b ohne Rest
:	sodass	z.B. $\forall a, b \in \mathbb{Z} : a \mid b$
mod	modulo	$a \equiv b \pmod{n} \iff n \mid (a - b)$, mit $a, b, n \in \mathbb{Z}$

Lizenz

- Unsere Folien sind frei!
- Jeder darf die Folien unter den Bedingungen der **GNU General Public License v3** (oder jeder späteren Version) weiterverwenden.
- Ihr findet den Quelltext unter `https://www.github.com/FIUS/theo-vorkurs-folien`

Online-Whiteboard