

Grundbegriffe der Informatik

Überblick

Mattias Ulbrich
(basierend auf Folien von Thomas Worsch)

KIT · Institut für Theoretische Informatik

Wintersemester 2023/2024

Kapitel 2: Signale, Nachrichten, Daten, Informationen

- Das sollten Sie mitnehmen:
 - Rechner verarbeiten
 - Zeichen/Nachrichten/Daten
 - **nicht** Informationen
 - nutzen dabei Signale (Übertragung) und Inschriften (Festhalten/Speichern)

- Das müssen Sie üben:
 - immer unterscheiden zwischen
 - dem, was hingeschrieben ist, also der Form (**Syntax**), und
 - dem, was es bedeutet (**Semantik**).

Kapitel 3: Mengen, Alphabete, Abbildungen

- Das sollten Sie mitnehmen:
 - *Alphabete*
 - *Mengenlehre*: u.a. Klassenterme, Kardinalitäten
 - *binäre Relationen*
 - *Abbildungen*
 - Spezialfälle: injektiv, surjektiv, bijektiv, partiell
 - Abbildungen (=Funktionen) sind auch binäre Relationen
 - partielle Abbildungen

- Das sollten Sie üben:
 - Benutzung der Begriffe Alphabet, Relation, Abbildung
 - „Rechnen“ mit Mengen
 - einfaches Argumentieren

Kapitel 4: Wörter

Das sollten Sie mitnehmen:

- ein *Wort* ist eine Folge von Symbolen
- eine *formale Sprache* ist eine Menge von Wörtern
- *induktive Definitionen*
erlauben, Pünktchen zu vermeiden ...

Das sollten Sie üben:

- Umgang mit formalen Definitionen
- „Rechnen“ mit Wörtern, insbesondere Konkatenation

Kapitel 5: Aussagenlogik

Das sollten Sie mitnehmen:

- *aussagenlogische Formeln* haben
 - Syntax
 - Semantik
- Formalisierung von *Beweisbarkeit* in Kalkülen
 - Zusammenhang mit Allgemeingültigkeit
 - Das Konzept einer *Inferenzregel*

Das sollten Sie üben:

- Rechnen mit booleschen Funktionen
- Überprüfen von Formeln auf Erfüllbarkeit und Allgemeingültigkeit
z.B. mit Wertetabelle
- Formal sauberes Beweisen (z.B. anhand der vorgestellten Muster)
- **Formalisieren von natürlich-sprachlichen Sätzen in Aussagenlogik**

Kapitel 6: Induktives Vorgehen

- Das sollten Sie mitnehmen:
 - *vollständige Induktion*
 - Methode, um etwas „für alle $n \in \mathbb{N}_0$ “ zu beweisen
 - Varianten, insbesondere „starke Induktion“
 - induktive Definitionen

Das sollten Sie üben:

- vollständige Induktionen
- induktive Definitionen

Kapitel 7: formale Sprachen

- Das sollten Sie mitnehmen:
 - was **formale Sprachen** sind,
 - wie ihr **Produkt** definiert ist und
 - wie **Konkatenationsabschluss** und ε -freier **Konkatenationsabschluss** definiert sind.
- Das sollten Sie üben:
 - Erkennen von Strukturen der Form L^* , L^+ , L_1L_2
 - Lesen von Ausdrücken der Form $(L_1^+L_2)^*$ usw.
 - „Rechnen“ mit formalen Sprachen

Einheit 8: Übersetzungen und Codierungen

- **Das sollten Sie mitnehmen:**
 - Umwandlungen zwischen Zahlen und Wörtern (Num_b , Repr_b)
 - Induktion ist ein „scharfes Schwert“ für Definitionen und Beweise
 - schon Leibniz kannte die Binärdarstellung
- **Das sollten Sie üben:**
 - selbe Zahlen verschieden repräsentieren
 - Definitionen auch in Randfällen ausprobieren
 - starke Induktion (aber das hatten wir ja schon ...)

- Das sollten Sie mitnehmen:
 - Übersetzungen sind in verschiedenen Situationen nützlich
 - Homomorphismen
 - $f : A \rightarrow B^*$ induziert Homomorphismus $f^{**} : A^* \rightarrow B^*$
 - Codes, präfixfrei
 - (UTF-8 als Beispiel)
- Das sollten Sie üben:
 - Homomorphismen anwenden
 - Mit Homomorphismen umgehen können
 - Decodieren

- **Das sollten Sie mitnehmen:**
 - Huffman-Codierung liefert kürzest mögliche präfixfreie Codes
 - „Algorithmus“ zur Bestimmung des Huffman-Baumes
- **Das sollten Sie üben:**
 - Huffman-Codes berechnen (Huffman-Bäume aufbauen)
 - Erkennen, ob/für welches Wort ein Code Huffman-Code ist

- Das sollten Sie mitnehmen:
 - Zweierkomplement-Darstellung
- Das sollten Sie üben:
 - selbst Zahlen verschieden repräsentieren und
 - damit operieren
 - Definitionen auch in Randfällen ausprobieren

Einheit 9: Speicher

- Das sollten Sie mitnehmen:
 - Bit und Byte / Octet
 - binäre Größenpräfixe
- Das sollten Sie üben:
 - Rechnen mit binären Größenpräfixen

■ Das sollten Sie mitnehmen:

- Abbildungen kann man sich gut als Tabelle vorstellen
- Abbildungen sind auch nur Werte (“Mengenwerte”)

Das sollten Sie üben:

- Abbildungen, die Abbildungen auf Abbildungen abbilden
- Modellierung mithilfe von Abbildungen und Zuweisungen an Abbildungen

Kapitel 10: Prozessor

- **Das sollten Sie mitnehmen:**
 - Grobstruktur
 - Steuerwerk, Rechenwerk, Speicherwerk
 - Maschinenbefehle
 - Mikroprogramme
- **Das sollten Sie üben:**
 - einfache ALU-Rechnungen
 - ganz einfache Maschinenprogramme

Kapitel 12: kontextfreie Grammatiken

- Das sollten Sie mitnehmen:
 - **Klammerstrukturen** sind wichtig.
 - Manchmal kann man sich einem **Fixpunkt** «annähern».
 - Fixpunkt von f ist ein x mit $x = f(x)$.
 - so kann man $L = \{\varepsilon\} \cup LL \cup (L)$ auch sehen ...
- Das sollten Sie üben:
 - keine Angst vor dem Lesen und Finden von Beweisen
 - in Ruhe hingucken:
passende Vorgehensweise drängt sich manchmal fast auf

- Das sollten Sie mitnehmen:
 - kontextfreie Grammatik
 - Ableitung
 - erzeugte formale Sprache
 - Ableitungsbaum
- Das sollten Sie üben:
 - (semi-)reale Produktionenmengen lesen (Java, ...)
 - zu formaler Sprache die erzeugende kontextfreie Grammatik konstruieren
 - zu kontextfreier Grammatik die erzeugte formale Sprache bestimmen

- Das sollten Sie mitnehmen:
 - Produkte und Potenzen von Relationen
 - reflexive und transitive Relationen
 - reflexiv-transitive Hülle einer Relation
 - «klassisches» Beispiel: Ableitbarkeit $\Rightarrow^*$
- Das sollten Sie üben:
 - Transitivität nachweisen
 - Bilder von Relationen malen

Kapitel 13: Prädikatenlogik erster Stufe

- Das sollten Sie mitnehmen:

Syntaktische Kategorien:

- Signatur
- Terme
- atomare Formeln
- Formeln

- Das sollten Sie üben:

- Formeln schreiben
 - Syntaxfehler vermeiden ; -)
 - Formalisieren von natürlich-sprachlichen Aussagen in Prädikatenlogik

- Das sollten Sie mitnehmen:
 - Interpretationen
 - Werte in D für jeden Term
 - Wahrheitswert für jede Formel
 - Modelle
- Das sollten Sie üben:
 - Terme und Formeln auswerten
 - Formeln auf Allgemeingültigkeit/Erfüllbarkeit prüfen
 - Modelle / Nicht-erfüllende Interpretationen angeben.

- Das sollten Sie mitnehmen:
 - informeller *Algorithmusbegriff*
 - *Schleifeninvarianten*
- Das sollten Sie üben:
 - Schleifeninvarianten finden
 - Wertetabellen können helfen
 - Korrektheitsbeweise aufstellen mit Hoare-Kalkül
(für Klausur: nur auf ganz kleinen Beispielen)

Kapitel 15: Graphen

- Das sollten Sie mitnehmen:
 - gerichtete Graphen drücken Beziehungen aus (**Relationen**)
 - Pfade
 - strenger Zusammenhang
 - Bäume
 - Zyklen, zyklensfrei
- Das sollten Sie üben:
 - Malen von Graphen
 - sehen, wann Graphen isomorph sind

- **Das sollten Sie mitnehmen:**
 - ungerichtete Graphen:
 - Unterschiede zu gerichteten Graphen
 - Gemeinsamkeiten mit gerichteten Graphen
 - Ungerichtete können in gerichtete überführt werden
- **Das sollten Sie üben:**
 - Benutzung der Begriffe
 - Malen von Graphen, hübsche und hässliche

- Das sollten Sie mitnehmen:

- Repräsentation von Relationen als Matrizen
- z. B. Kantenrelation eines Graphen: Adjazenzmatrix
- Wegematrix

- Das sollten Sie üben:

- zu gegebenem Graphen die Adjazenzmatrix hinschreiben
- zu gegebener Adjazenzmatrix den Graphen hinmalen
- z. B. für irgendwelche «speziellen» Graphen und Matrizen
(wie sehen Matrizen für Graphen/Relation mit bestimmter Eigenschaft aus)

Kapitel 16: Erste Algorithmen in Graphen

- **Das sollten Sie mitnehmen:**
 - Repräsentation von Relationen als Matrizen
 - z. B. Kantenrelation eines Graphen: Adjazenzmatrix
- **Das sollten Sie üben:**
 - zu gegebenem Graphen die Adjazenzmatrix hinschreiben
 - zu gegebener Adjazenzmatrix den Graphen hinmalen
 - z. B. für irgendwelche „speziellen“ Graphen und Matrizen

- Das sollten Sie mitnehmen:
 - die naheliegende Idee nicht immer die beste
 - Denken, Mathematik, Kreativität, Einfach-mal-drüber-schlafen, ...
- Das sollten Sie üben:
 - Aufwandsabschätzungen bei (ineinander geschachtelten) Schleifen
 - auch mal verrückte Ideen ausprobieren

Einheit 18: Quantitative Aspekte von Algorithmen

- Das sollten Sie mitnehmen:
 - Bedarf an
 - Rechenzeit und
 - Speicherplatzbedarf
 - wichtige **Komplexitätsmaße**
 - oft Quantifizierung in Abhängigkeit von Problemgröße
 - oft schlimmster Fall (**worst case**)
 - gelegentlich ein «mittlerer» Fall (average case)
- Das sollten Sie üben:
 - Abschätzen/ausrechnen wie oft ein Programmstück, z. B. ein Schleifenrumpf, durchlaufen wird.

- Das sollten Sie mitnehmen:
 - Definitionen von $O(f)$, $\Theta(f)$, $\Omega(f)$
 - Definitionen von $\leq$, $\asymp$, $\geq$
- Das sollten Sie üben:
 - Anschauung für $O(f)$, $\Theta(f)$, $\Omega(f)$
 - rechnen mit $O(f)$, $\Theta(f)$, $\Omega(f)$

- **Das sollten Sie mitnehmen:**
 - bei algorithmischen Problemen kann man überraschende Dinge tun
 - teile und herrsche
 - Rekursionsformel für Abschätzung von (z. B.) Laufzeiten
- **Das sollten Sie üben:**
 - in dieser Vorlesung: rechnen mit $O(\cdot)$, $\Theta(\cdot)$, $\Omega(\cdot)$
 - spätestens in kommenden Semestern: rekursive Algorithmen

Kapitel 14: Endliche Automaten

- Das sollten Sie mitnehmen:
 - alles endlich
 - alles endlich beschreibbar
 - im Beispiel Vokabelkasten:
 - aktueller Zustand und aktuelle Eingabe legen eindeutig nächsten Zustand fest
 - aktueller Zustand legt Ausgabe fest
- Das sollten Sie üben:
 - Zustandsdiagramm lesen

- Das sollten Sie mitnehmen:
 - «offizielle» Definition von Moore-Automat:
 - Z
 - $z_0 \in Z$
 - X
 - $f : Z \times X \rightarrow Z$
 - Y
 - $h : Z \rightarrow Y^*$ Ausgabe hängt nur vom Zustand ab
- Das sollten Sie üben:
 - vorgegebenes Verhalten mit Moore-Automat realisieren
 - vorgegebenen Moore-Automaten analysieren

- Das sollten Sie mitnehmen:
 - «offizielle» Definition endlicher Akzeptoren:
 - Z
 - $z_0 \in Z$
 - X
 - $f : Z \times X \rightarrow Z$
 - $F \subseteq Z$
 - Wenn ein «sehr langes» Wort akzeptiert wird,
 - dann läuft der Automat in einer Schleife,
 - die beliebig oft durchlaufen werden kann ohne Akzeptanz zu ändern
- Das sollten Sie üben:
 - gegeben L : konstruiere A mit $L(A) = L$
 - gegeben A : bestimme $L(A)$

Kapitel 15: Reguläre Ausdrücke und rechtslineare Grammatiken

- Das sollten Sie mitnehmen:
 - Definition «klassischer» regulärer Ausdrücke
 - atomare: $\emptyset, a \in A$
 - zusammengesetzte: $(R_1 \mid R_2), (R_1 R_2), (R)^*$
 - wissen: reguläre Ausdrücke und die Verallgemeinerung **Regular Expressions** z. B. bei Textverarbeitungsaufgaben manchmal nützlich
- Das sollten Sie üben:
 - zu L ein R mit $L(R) = L$ finden
 - zu R das $L(R)$ finden

- Das sollten Sie mitnehmen:
 - Definition **rechtslineare Grammatiken**
- Das sollten Sie üben:
 - rechtslineare Grammatiken konstruieren
(zu gegebenem Akzeptor, regulären Ausdruck,
formaler Sprache)

Das ist neu!

- **Das sollten Sie mitnehmen:**
 - Das induktive Prinzip von $\mathbb{N}_0$ kann verallgemeinert werden.
 - Induktive Strukturen werden aus Basisfällen und Konstruktoren generiert (über die möglichen Terme)
 - Kantorowitsch-Bäume stellen Elemente induktiver Strukturen als Baum dar.
 - strukturelle Induktion “kann nicht mehr”, ist auf starke vollständige Induktion reduzierbar.
 - Viele Strukturen aus der Vorlesung sind induktiv.
- **Das sollten Sie üben:**
 - Aussagen über induktiven Strukturen mit Hilfe von struktureller Induktion beweisen.

Kapitel 16/16a: Turingmaschinen

- Das sollten Sie mitnehmen:
 - Turingmaschinen
 - Steuereinheit endlich
 - Band unendlich, aber
nur endlich viel «nicht leer»
 - alles endlich beschreibbar
 - klassische Formalisierung von «Algorithmus»
 - Berechnungen
 - haltende
 - nicht haltende
- Das sollten Sie üben:
 - Beispielturingmaschinen konstruieren
 - Beispielturingmaschinen verstehen
 - Ablaufprotokolle von Turingmaschinen erstellen

- Das sollten Sie mitnehmen:
 - **Zeitkomplexität** und **Raumkomplexität**
 - in Abhängigkeit von Eingabegröße der schlimmste Fall
 - evtl. nur obere Schranke
 - **Komplexitätsklassen**
 - durch **Beschränkung der zur Verfügung stehen Ressourcen**, also
 - z. B. Schranken für Zeitkomplexität oder/und Raumkomplexität
 - wichtig: **P** und **PSPACE**
- Das sollten Sie üben:
 - Abschätzung der Zeit- und Raumkomplexität von TM

- Das sollten Sie mitnehmen:
 - Das Halteproblem ist unentscheidbar.
 - viele andere interessierende Probleme auch
 - **Diagonalisierung** zeigt, dass etwas nicht existieren kann.
 - Busy-Beaver-Funktion wächst schneller als jede berechenbare Funktion.
- Das sollten Sie üben:
 - informelle algorithmische Beschreibungen in Turingmaschinen überführen

Kapitel 20: Relationen

- Das sollten Sie mitnehmen:
 - Äquivalenzrelationen
 - Beispiel Kongruenz modulo n
- Das sollten Sie üben:
 - definierenden Eigenschaften überprüfen
 - Anzahl Äquivalenzklassen bestimmen

- Das sollten Sie mitnehmen:
 - Kongruenzrelationen: Verträglichkeit
 - induzierte Abbildungen/Operationen für Äquivalenzklassen
- Das sollten Sie üben:
 - mit Äquivalenzklassen rechnen

- Das sollten Sie mitnehmen:
 - Halbordnungen sind
 - reflexiv,
 - antisymmetrisch und
 - transitiv
- Das sollten Sie üben:
 - Nachweis der Eigenschaften von Halbordnungen
 - Beweise einfacher Aussagen
 - an ungewohnte Eigenschaften von Halbordnungen gewöhnen
 - Unendlichkeit lässt grüßen

- **Das sollten Sie mitnehmen:**
 - totale Ordnungen sind
 - Halbordnungen
 - ohne unvergleichbare Elemente
 - Anwendung an diversen Stellen in der Informatik
 - z. B. Semantik, Testmuster, ...
- **Das sollten Sie üben:**
 - Nachweis der Eigenschaften von totalen Ordnungen
 - Beweise einfacher Aussagen
 - an ungewohnte Eigenschaften von Ordnungen gewöhnen (Unendlichkeit ...)