

# Themenüberblick

Dies ist eine Art Checkliste für die Klausurvorbereitung. Zu jedem Thema im Skript habe ich ein paar Leitfragen aufgelistet und die ganzen Übungsaufgaben in den 13 Blättern + Altklausuren nach entsprechenden Themen sortiert. Dabei sind die in Klammern gesetzten Aufgaben, meiner bescheidenen Meinung nach, entweder irrelevant oder zu schwer für eine Klausur. Wie immer, alle Angaben ohne Gewähr :-)

## Bemerkungen:

- Die Klausuren von vor 2008 habe ich ignoriert, weil die Vorlesung damals von anderen Professoren gehalten wurde und deswegen vom Stoff her ein bisschen anders ist. 2008 bis 2012 sollte aber zum Üben reichen ;-)
- Die Wiederholungsklausuren von 2008 und 2009 habe ich leider nirgends gefunden.
- Zum Gödelisieren der Aufgaben habe ich folgende Abkürzungen benutzt:

B: Blatt	H: Hausaufgabe
M: Midtermklausur	V: Vorbereitungsaufgabe
E: Endtermklausur	T: Tutoraufgabe
S: Semestralklausur	Z: Zusatzaufgabe
W: Wiederholungsklausur	A: Aufgabe

- Folgende Aufgaben bestehen aus mehreren (Multiple-Choice-) Fragen zu verschiedenen Themen: B3V1, B4H1, B4V2, B5V1, B6V1, B7V1, B9H1, B9V2, B11H1, B11T2, B12H1, B12H3, B12V1, B13H3, W10A6, W11A6.1 W11.2 + natürlich Aufgabe 1 bei allen Klausuren.

## 1 Grundbegriffe

(s. Überblicksfolien 01)

- Was sind formale Sprachen und welche Operationen gibt es auf diese?  
→ Aufgaben: B1V1, (B1T1), (B2H1), (B2H2), (B2H3)

## 2 Reguläre Sprachen

### 2.1 Deterministische endliche Automaten (DFAs)

(s. Überblicksfolien 02)

- Wie sind DFAs definiert und wie stellt man sie graphisch dar?  
→ Aufgaben: B1T2, B1T3.1, (B3H1)
- Wie sind reguläre Sprachen definiert?

### 2.2 Nichtdeterministische endliche Automaten (NFAs)

(s. Überblicksfolien 03)

- Wie sind NFAs definiert und wie stellt man sie graphisch dar?
- Wieso kann man DFAs als Spezialfall von NFAs betrachten, obwohl formal kein DFA ein NFA sein kann?  
→ Aufgaben: B1V2, B1TA3.2

## 2.3 Äquivalenz von NFAs und DFAs

(s. Überblicksfolien 07)

- Wie kann man aus jedem NFA einen DFA konstruieren, der dieselbe Sprache akzeptiert? (Potenzmengenverfahren)  
→ Aufgaben: B2T2.3, B3H2, B4H2.1, M09A2, W10A2, S12A2.1

## 2.4 NFAs mit $\epsilon$ -Übergängen ( $\epsilon$ -NFAs)

(s. Überblicksfolien 03 und 06)

- Wie sind  $\epsilon$ -NFAs definiert und wie stellt man sie graphisch dar?  
→ Aufgaben: (B3H4)
- Wieso kann man NFAs als Spezialfall von  $\epsilon$ -NFAs betrachten, obwohl formal kein NFA ein  $\epsilon$ -NFA sein kann?
- Wie kann man aus jedem  $\epsilon$ -NFA einen NFA konstruieren, der dieselbe Sprache akzeptiert? (Sättigungsalgorithmus)  
→ Aufgaben: B2T2.2

## 2.5 Reguläre Ausdrücke (REs)

(s. Überblicksfolien 04 und 05)

- Wie sind REs definiert?  
→ Aufgaben: B2V1, M10A4.2, S12A3.1, M11A2.1, M11A2.2
- Wann sind zwei REs äquivalent?  
→ Aufgaben: (B2T1), (W10A4.1)
- Wie kann man aus jedem RE einen  $\epsilon$ -NFAs konstruieren, der dieselbe Sprache akzeptiert? (Satz von Kleene)  
→ Aufgaben: B2T2.1, W11A2.1
- Wieso ist jede endliche Sprache regulär?

## 2.6 Abschlusseigenschaften regulärer Sprachen

- Unter welchen Operationen sind reguläre Sprachen abgeschlossen?
- Seien  $L, L_1, L_2$  reguläre Sprachen. Wie zeigt man, dass die Sprachen  $L_1L_2, L_1 \cup L_2, L^*, \bar{L}, L_1 \cap L_2, L_1 \setminus L_2$  und  $L^R$  auch regulär sind?  
→ Aufgaben: M09A3.1, B4H2.1
- Wie kann man diese Eigenschaften benutzen um die Regularität/Nichtregularität einer Sprache zu zeigen?  
→ Aufgaben: M09A5.2
- Wie kann man die Regularität einer komplexeren Sprache zeigen?  
→ Aufgaben: B2T3, B4H3, B4H4, B5H1, M10A3, M09A4, M08A4, W11A3, W10A4.2, M11A2.3

## 2.7 Rechnen mit regulären Ausdrücken

- Welche Rechenregeln gibt es für REs?

## 2.8 Pumping Lemma für reguläre Sprachen

(s. Überblicksfolien 08)

- Wieso gilt das (reguläre) Pumping Lemma für alle reguläre Sprachen?  
→ Aufgaben: B3H3
- Wie kann man es benutzen, um zu beweisen, dass eine Sprache nicht regulär ist?  
→ Aufgaben: B3T1, B5H2, B6H1, W10A3.1, W11A2.3
- Kann man mit dem Pumping Lemma beweisen, dass eine Sprache doch regulär ist?  
→ Aufgaben: M09A5.1

## 2.9 Entscheidbarkeit

- Welche Probleme für reguläre Sprachen sind entscheidbar?  
→ Aufgaben: B7H1
- In welcher Laufzeit?

## 2.10 Äquivalenz regulärer Ausdrücke

- Was darf man machen, um Beweise über die Äquivalenz von regulären Ausdrücken zu vereinfachen?

## 2.11 Automaten und Gleichungssysteme

(s. Überblicksfolien 09)

- Was besagt das Ardens Lemma und welche Bedingungen müssen erfüllt sein?  
→ Aufgaben: (B3T2)
- Wie kann man aus jedem NFA (oder DFA) einen RE konstruieren, der dieselbe Sprache akzeptiert? (Gleichungssysteme + Ardens Lemma)  
→ Aufgaben: B3T3, B5H3, M08A5, M11A3

## 2.12 Minimierung endlicher Automaten

(s. Überblicksfolien 10)

- Wie ist Äquivalenz von Zuständen definiert, was ist der Quotientenautomat zu einem DFA und wie kann man ihn konstruieren?  
→ Aufgaben: B4V1, B4T2, (B5H4), B6H3, M10A2, M09A3.2, M08A3, S12A2.2, M11A4.1
- Wie ist die Äquivalenz auf Wörtern definiert, was ist der kanonische Minimalautomat zu einer Sprache und wie kann man ihn konstruieren?  
→ Aufgaben: B4T1, M11A4.2
- Wie kann man mit ihm die Regularität/Nichtregularität einer Sprache beweisen? (Satz von Myhill-Nerode)  
→ Aufgaben: (M09A5.2)

## 3 Kontextfreie Sprachen

### 3.1 Kontextfreie Grammatiken (CFGs)

(s. Überblicksfolien 11)

- Wie sind CFGs definiert und wie können sie dargestellt werden?  
→ Aufgaben: B4T3, W10A3.2, W11A4
- Wie sind Kontextfreie Sprachen (CFLs) definiert?
- Was sind links- und rechts-lineare CFGs?  
→ Aufgaben: B6H4, W11A2.2
- Was ist ein (unendlicher) Ableitungsbaum ( $\neq$  Syntaxbaum)?

### 3.2 Induktive Definitionen, Syntaxbäume und Ableitungen

- Sei  $P(w)$  ein einstelliges Prädikat, also eine Eigenschaft von  $w \in \Sigma^*$ . Wie beweist man  $w \in L(S) \Rightarrow P(w)$  (Induktion über die Erzeugung von  $w$ ) und  $w \in L(S) \Leftarrow P(w)$  (meistens mit vollständiger Induktion über  $|w|$ )?  
Aufgaben: B5T1, B5T3, B6H2, B7H2, B8H2.2, M08A2, M09A6.1, M09A6.2, M10A5.2, W10A3.3, M11A5.1, W11A4.2
- Was ist ein Syntaxbaum und wann ist eine CFG mehrdeutig bzw. eine CFL inhärent mehrdeutig?  
→ Aufgaben: (B5T2), B7H3, S124.1, S12A4.1

### 3.3 Die Chomsky-Normalform (CNF)

(s. Überblicksfolien 14)

- Wann ist eine CFG in CNF?
- Wie kann man aus einer beliebigen CFG eine äquivalente CFG in CNF konstruieren?  
→ Aufgaben: (B6H5), S12A4.2, M09A6.3

### 3.4 Das Pumping-Lemma für kontextfreie Sprachen

(s. Überblicksfolien 12)

- Wieso gilt das (kontextfreie) Pumping Lemma für alle kontextfreie Sprachen?
- Wie kann man es benutzen, um zu beweisen, dass eine Sprache nicht kontextfrei ist?  
→ Aufgaben: B6T1.1, (B7H4), B8H1, M11A5.2, S12A3.2
- Kann man mit dem Pumping Lemma beweisen, dass eine Sprache doch kontextfrei ist?

### 3.5 Abschlusseigenschaften kontextfreier Sprachen

- Unter welchen Operationen sind kontextfreie Sprachen abgeschlossen?
- Seien  $L, L_1, L_2$  kontextfreie Sprachen. Wie zeigt man, dass die Sprachen  $L_1L_2, L_1 \cup L_2, L^*$  und  $L^R$  auch kontextfrei sind? (Achtung:  $L_1 \cap L_2$  ist nicht notwendigerweise kontextfrei!)  
→ Aufgaben: B10H1.1
- Wie kann man diese Eigenschaften benutzen um die Kontextfreiheit/Nichtkontextfreiheit einer Sprache zu zeigen? (Analog zu den regulären Sprachen)

### 3.6 Algorithmen für kontextfreie Grammatiken

- Was sind nützliche, erzeugende und erreichbare Symbole und wie kann man eine Grammatik konstruieren, die nur nützlichen Symbole enthält?  
→ Aufgaben: B6H6

### 3.7 Der Cocke-Younger-Kasami-Algorithmus (CYK)

(s. Überblicksfolien 13)

- Wie (und wieso?) funktioniert der CYK-Algorithmus und was kann man ihm machen?  
→ Aufgaben: B6T2, B8H2.1, M10A5.1, S12A4.2, W11A4.3

### 3.8 Kellerautomaten (PDAs)

(s. Überblicksfolien 15)

- Wie sind PDAs definiert, wie stellt man sie graphisch dar und was ist eine Konfiguration eines PDAs?  
→ Aufgaben: B6T3, (B8H3), E09A2, (E08A2)
- Wie kann man aus jeder CFG einen PDA konstruieren, der dieselbe Sprache akzeptiert?

### 3.9 Tabellarischer Überblick

- Das kommt direkt in die Formelsammlung! ;-) (Achtung: DCFLs und DPDAS sind deterministische CFGs bzw. PDAs und wurden nicht in der Vorlesung behandelt)

### 3.10 Die Chomsky-Hierarchie

- Welche sind die Typ-3-, Typ-2- und Typ-0-Sprachen und wieso gilt  $\text{Typ } 3 \subset \text{Typ } 2 \subset \text{Typ } 0$ ? (Typ-1-Sprachen wurden zwar in der Vorlesung definiert, aber sie wurden kaum geübt)  
→ Aufgaben: (B10H1.2)

## 4 Berechenbarkeit und Entscheidbarkeit

### 4.1 Der Begriff der Berechenbarkeit

- Wann ist eine Funktion intuitiv berechenbar und was heißt das für uns? (Church-Turing These)

### 4.2 Turingmaschinen (TMs)

(s. Überblicksfolien 16)

- Wie sind TMs definiert, wie stellt man sie graphisch dar und was ist eine Konfiguration einer TM?  
→ Aufgaben: B7V2, B7T1, B8H4, B9H2, (B9H4), E11A6, (E10A5), W11A7, W10A7, S12A5
- Was ist der Unterschied zwischen einer deterministischen und einer nichtdeterministischen TM?
- Wann ist eine Funktion Turing-berechenbar?
- Welche Sprachen werden von TMs akzeptiert?

### 4.3 Programmieren von $k$ -Band-TMs

- Wie funktionieren  $k$ -Band-TMs?
- Welche Übersetzungen kennen wir zwischen  $k$ -Band- und normale TMs?  
→ Aufgaben: (B8T1)

#### 4.4 LOOP-, WHILE- und GOTO-Berechenbarkeit

(s. Überblicksfolien 17)

- Wie sind LOOP-Programme definiert und wann ist eine Funktion LOOP-berechenbar?  
→ Aufgaben: (B7T2), B8T3.1, B9H3
- Welche syntaktische Abkürzungen könnte man als LOOP-berechenbar annehmen?
- Wie sind WHILE-Programme definiert und wann ist eine Funktion WHILE-berechenbar?  
→ Aufgaben: B10H4.3
- Wie sind GOTO-Programme definiert und wann ist eine Funktion GOTO-berechenbar?
- Welche Übersetzungen kennen wir zwischen  $k$ -Band-TMs, GOTO-, WHILE- und LOOP-Programmen?

#### 4.5 Primitiv rekursive Funktionen (PR Funktionen)

(s. Überblicksfolien 18)

- Wie sind PR Funktionen definiert und was versteht man unter der erweiterten Komposition und dem erweiterten Schema der primitiven Rekursion?  
→ Aufgaben: B8V1, B8T2, B9T1.1, (B10H2), B10H3, B11H2, (B13Z1), E11A2, E10A2, E09A4, E08A4, W11A5, W10A5, S12A7
- Welche Funktionen und welche syntaktische Abkürzungen könnte man als PR-berechenbar annehmen?
- Wann ist ein Prädikat  $P(x)$  PR?

#### 4.6 PR = LOOP

- Wie ist die Cantorsche Paarungsfunktion  $c$  und die Umkehrfunktionen  $p_1$  und  $p_2$  definiert und wie kann man damit Zahlenpaare kodieren?  
→ Aufgaben: B8T3.2
- Wie kann man mit  $\langle \rangle$  und  $d_0, d_1, \dots, d_k$  beliebige Tupeln kodieren?  
→ Aufgaben: B10H4.1
- Welche Übersetzungen kennen wir zwischen PR und LOOP?

#### 4.7 Die $\mu$ -rekursiven Funktionen ( $\mu$ R)

- Wie sind  $\mu$ R Funktionen definiert?  
→ Aufgaben: B9T1.2
- Welche Übersetzungen kennen wir zwischen  $\mu$ R und WHILE?

#### 4.8 Die Ackermann-Funktion

- Wie ist die Ackermann-Funktion definiert?  
→ Aufgaben: B9V1
- Welche Eigenschaften besitzt sie?  
→ Aufgaben: (B11H4)

## 4.9 Entscheidbarkeit und das Halteproblem

- Wann ist eine Menge/Sprache bzw. ein Prädikat entscheidbar/unentscheidbar?  
→ Aufgaben: B3H5, B10H4.2, B12H4.3, B13H2.1, (B13H2.2), E10A3.1, ET11A3.2, E09A3.3, E11A3.2
- Was bedeutet  $M_w$  bzw.  $\varphi_w$  und was ist die Gödelisierung eines Objekts?  
→ Aufgaben: B10V1, B10T1
- Was ist eine Universelle TM?
- Was bedeutet  $M_w[x] \downarrow$ , wie sind die Halteprobleme  $K$ ,  $H$  und  $H_0$  definiert und welche Eigenschaften besitzen sie?
- Wann ist eine Menge/Sprache auf eine andere reduzierbar und was bedeutet dies für die Entscheidbarkeit der beiden?  
→ Aufgaben: B9T2, B12H2, B12H4.2, E08A5.3, E09A3.2

## 4.10 Semi-Entscheidbarkeit

- Wann ist eine Sprache bzw. ein Prädikat semi-entscheidbar?  
→ Aufgaben: E08A5.1, E11A3.3
- Wann ist eine Sprache bzw. ein Prädikat rekursiv aufzählbar?  
→ Aufgaben: B9T3, B11H3
- Was ist zur semi-entscheidbarkeit äquivalent?
- Welche Sprachen sind semi-entscheidbar? Welche nicht?

## 4.11 Die Sätze von Rice und Shapiro

- Was besagt der Satz von Rice und was kann man damit zeigen?  
→ Aufgaben: B10T2, B12H4.1, E08A5.2, E09A3.1, E10A3.2, E11A3.1, W11A6
- Was besagt der Satz von Rice-Shapiro und was kann man damit zeigen?

## 4.12 Das Postsche Korrespondenzproblem (PCP)

- Wie ist das PCP definiert und welche Eigenschaften hat es?  
→ Aufgaben: (B10T3), B11T1, B12H5, E10A4, E11A5

## 4.13 Unentscheidbare CFG-Probleme

- Welche Probleme für CFGs sind unentscheidbar?

# 5 Komplexitätstheorie

## 5.1 Die Komplexitätsklasse $\mathcal{P}$

- Wie sind  $time_M$ ,  $TIME(f(n))$  und  $\mathcal{P}$  definiert?
- Wie zeigt man, dass eine Sprache  $L$  in  $\mathcal{P}$  liegt?  
→ Aufgaben: B12V2
- Wie sind folgende Probleme aus  $\mathcal{P}$  definiert?
  - EULER
  - 2COL

## 5.2 Die Komplexitätsklasse $\mathcal{NP}$

- Wie sind  $ntime_M$ ,  $NTIME(f(n))$  und  $\mathcal{NP}$  definiert?
- Wieso ist  $\mathcal{P}$  eine Teilklasse von  $\mathcal{NP}$ ?
- Welche äquivalente Beschreibung von  $\mathcal{NP}$  gibt es? (Zertifikat + Verifikator)
- Wie kann man zeigen, dass eine Sprache  $L$  in  $\mathcal{NP}$  liegt?  
→ Aufgaben: B12T1
- Was heißt LOOP-entscheidbar?

## 5.3 NP-Vollständigkeit

- Wie ist die polynomielle Reduzierbarkeit definiert?  
→ Aufgaben: B11V1
- Wann ist ein Problem  $\mathcal{NP}$ -hart und wann  $\mathcal{NP}$ -vollständig?
- Wie zeigt man, dass ein Problem  $\mathcal{NP}$ -hart oder sogar  $\mathcal{NP}$ -vollständig ist?  
→ Aufgaben: B12.T2, B13H5, B13H6, B13H7, B13H8, B13Z2, E08A3, E09A5, E10A6, E11A4, W11A8, S12A6
- Was besagt der Satz von Cook und warum ist er so wichtig für die Klasse der  $\mathcal{NP}$ -vollständigen Probleme?
- Wie sind die  $\mathcal{NP}$ -vollständige Probleme SAT und 3COL definiert?

## 5.4 Weitere $\mathcal{NP}$ -vollständige Probleme

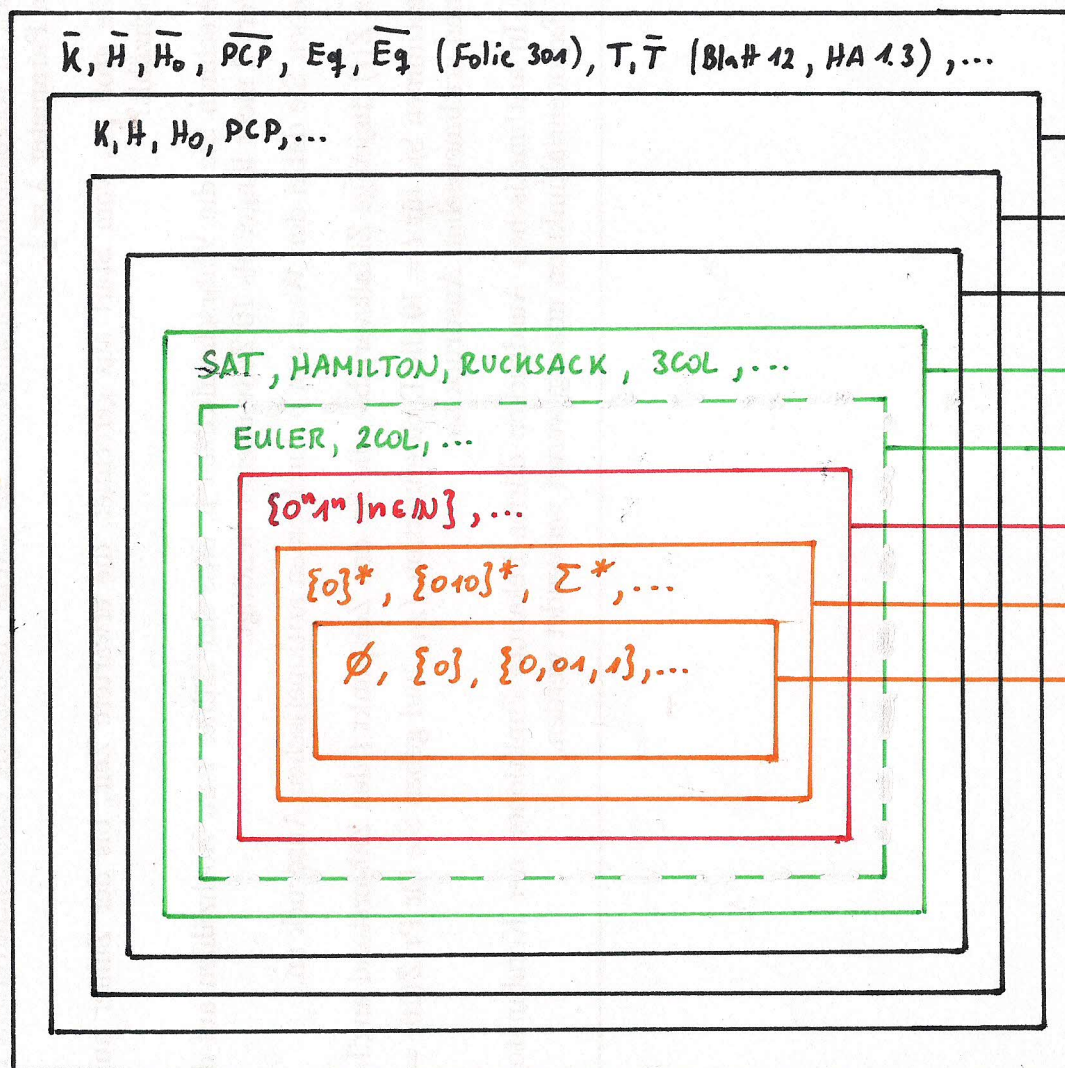
- Wie sind folgende  $\mathcal{NP}$ -vollständige Probleme definiert?
  - KNF-SAT
  - 3KNF-SAT
  - HAMILTON
  - RUCKSACK
  - FÄRBBARKEIT
  - MENGENÜBERDECKUNG
  - CLIQUE
  - PARTITION
  - BIN PACKING
  - TRAVELLING SALESMAN

## 5.5 Die Unvollständigkeit der Arithmetik

- Was hat Gödel gemacht außer Sachen zu gödelisieren?



# Bild 1



• alle Sprachen über  $\Sigma^*$

• semi-entscheidbar (Typ 0)

= rekursiv aufzählbar

• entscheidbar  
(unentsch. = nicht entsch.)

• LOOP-entscheidbar

• NP (Beispiele alle NP-vollst.)

• P (P-NP-Grenze offen!)

• Kontextfrei (Typ 2)

• regulär (Typ 3)

• endlich

Teil 3

Teil 4

Teil 2

Teil 1

## Maschinenmodelle:

• Typ 3: DFA, NFA,  $\epsilon$ -NFA, RE, r-CFG, l-CFG

• Typ 2: GFG, PDA

• Typ 0: TM (egal welche Art), WHILE, GOTO,  $\mu R$

Bild 2

