

# Formale Systeme, Automaten, Prozesse

Vorlesung im Sommersemester 2014

---

Prof. Martin Grohe

Lehrstuhl Informatik 7 — Logik und Theorie Diskreter Systeme

# Organisatorisches

## Dozent



Prof. Dr. Martin Grohe

## Assistenten



Cornelius Aschermann



Erkal Selman



Sarah Winter

## Tutoren

Oliver Scheel, Erik Mühmer, Patrick Schmidt, Tim Quatmann, Patrick Julian Landwehr, Alexandra Angelika Grub, Erwin Fang, Fabian Philipp Schneider, Matthias Voit, Eva Fluck, Clara Elisabeth Scherbaum, Lukas Westhofen, Christoph Welzel, Jan Bruckner, Niklas Rieken, Russ Lucas Jukic, Daniel Esser

Zwei Registrierungen erforderlich:

- ▶ für die Vorlesung:

Im CampusOffice die Vorlesung auswählen

~> auf Zum klassischen Anmeldeverfahren klicken.

Dadurch Zugang zum L<sup>2</sup>P-Lernraum.

Alle Ankündigungen und das gesamte Material finden Sie dort.

- ▶ für die Übungsgruppe:

<https://aprove.informatik.rwth-aachen.de/fosap14/>

Dort können Sie Prioritäten für verschiedene Termine verteilen.

Achten Sie auf die Ordnung der Prioritäten (die größte Zahl ist die Übung, die sie am liebsten haben wollen)!

**Frist:** bis nächsten Montag (15.4.) um 12:00 Uhr

- ▶ **Globalübung:** Donnerstags 12:15 – 13:45, Beginn am 24. April 2014
- ▶ **Tutorien:** 17 Termine Dienstags und Mittwochs in unterschiedlichen Räumen zu unterschiedlichen Zeiten (Beginn 15. April)

Es gibt insgesamt 12 Übungsblätter, die für die Zulassung zur Klausur bearbeitet werden müssen.

Es gibt insgesamt 12 Übungsblätter, die für die Zulassung zur Klausur bearbeitet werden müssen.

- ▶ Ausgabe Übungsblätter immer Dienstags im L<sup>2</sup>P

Es gibt insgesamt 12 Übungsblätter, die für die Zulassung zur Klausur bearbeitet werden müssen.

- ▶ Ausgabe Übungsblätter immer Dienstags im L<sup>2</sup>P
- ▶ Zwei Aufgabentypen: Tutoraufgaben und Hausaufgaben



Es gibt insgesamt 12 Übungsblätter, die für die Zulassung zur Klausur bearbeitet werden müssen.

- ▶ Ausgabe Übungsblätter immer Dienstags im L<sup>2</sup>P
- ▶ Zwei Aufgabentypen: Tutoraufgaben und Hausaufgaben
- ▶ Bearbeitung (und Abgabe) der Hausaufgaben in Gruppen von je drei Studierenden; dazu eine Woche Zeit

Es gibt insgesamt 12 Übungsblätter, die für die Zulassung zur Klausur bearbeitet werden müssen.

- ▶ Ausgabe Übungsblätter immer Dienstags im L<sup>2</sup>P
- ▶ Zwei Aufgabentypen: **Tutoraufgaben** und **Hausaufgaben**
- ▶ Bearbeitung (und Abgabe) der Hausaufgaben in Gruppen von je drei Studierenden; dazu eine Woche Zeit
- ▶ Abgabe der Hausaufgaben: im Tutorium oder **zur Not** bis Mittwoch, 18:00 Uhr in den Kasten im Flur Lehrstuhl i7

Mit Übungsgruppe, Name und Matrikelnummer beschriften:

Gruppe: 12  
FoSAP-Übungsblatt 5

Erika Mustermann, 294255  
Otto Normalstudent, 315450

Bitte die Blätter zusammenheften!

Es gibt insgesamt 12 Übungsblätter, die für die Zulassung zur Klausur bearbeitet werden müssen.

- ▶ Ausgabe Übungsblätter immer Dienstags im L<sup>2</sup>P
- ▶ Zwei Aufgabentypen: **Tutoraufgaben** und **Hausaufgaben**
- ▶ Bearbeitung (und Abgabe) der Hausaufgaben in Gruppen von je drei Studierenden; dazu eine Woche Zeit
- ▶ Abgabe der Hausaufgaben: im Tutorium oder **zur Not** bis Mittwoch, 18:00 Uhr in den Kasten im Flur Lehrstuhl i7

Mit Übungsgruppe, Name und Matrikelnummer beschriften:

Gruppe: 12  
FoSAP-Übungsblatt 5

Erika Mustermann, 294255  
Otto Normalstudent, 315450

Bitte die Blätter zusammenheften!

- ▶ Donnerstags in der Globalübung wird eine Musterlösung der Hausaufgaben vorgestellt

Es gibt insgesamt 12 Übungsblätter, die für die Zulassung zur Klausur bearbeitet werden müssen.

- ▶ Ausgabe Übungsblätter immer Dienstags im L<sup>2</sup>P
- ▶ Zwei Aufgabentypen: **Tutoraufgaben** und **Hausaufgaben**
- ▶ Bearbeitung (und Abgabe) der Hausaufgaben in Gruppen von je drei Studierenden; dazu eine Woche Zeit
- ▶ Abgabe der Hausaufgaben: im Tutorium oder **zur Not** bis Mittwoch, 18:00 Uhr in den Kasten im Flur Lehrstuhl i7

Mit Übungsgruppe, Name und Matrikelnummer beschriften:

Gruppe: 12  
FoSAP-Übungsblatt 5

Erika Mustermann, 294255  
Otto Normalstudent, 315450

Bitte die Blätter zusammenheften!

- ▶ Donnerstags in der Globalübung wird eine Musterlösung der Hausaufgaben vorgestellt
- ▶ Rückgabe der korrigierten Hausaufgaben in den Tutorien

- ▶ Test unter Klausurbedingung am 5. Juni 2014
- ▶ Genaue Uhrzeit und Hörsäle werden noch bekanntgegeben
- ▶ Alle Informationen auch im L<sup>2</sup>P

## ► Zulassung zur Klausur:

- 50% der Punkte in den Hausaufgaben bis inklusive Übung 6
- 50% der Punkte in der Präsenzübung
- 50% der Punkte in den Hausaufgaben ab Übung 7

Falls Sie in der Präsenzübung mindestens 33%, aber weniger als 50% erreichen, können Sie das ausgleichen durch entsprechend mehr Punkte in den Hausaufgaben ab Übung 7 (Summe  $\geq 100\%$ ).

## ► Klausur:

- Klausur (PT1) am Di., 12.08.2014 nachmittags
- Nachholklausur (PT2) am Do., 18.09.2014 vormittags

## ► Alle Informationen auch im L<sup>2</sup>P

# Was tun bei Fragen ?

1. Erster Ansprechpartner ist der Tutor der Kleingruppe.
2. Sollte der Tutor die Frage nicht klären können, so können sie mich nach der Vorlesung oder einen Assistenten nach der Übung am Donnerstag ansprechen.
3. Dann erst, oder in dringenden(!) Fällen Email an:

`fosap@automata.rwth-aachen.de`

Bitte keine Email direkt an mich oder die Assistenten.

# Wie können Sie mich erreichen ?

1. Immer nach der Vorlesung.
2. In meiner Sprechstunde: Montags, 16:00-17:00.
3. Zur Not einen anderen Termin mit meiner Sekretärin vereinbaren.

Bitte keine Email an mich und keine Telefonanrufe.



- ▶ Hopcroft, John E. / Motwani, Rajeev / Ullman, Jeffrey D. (2011, 3. Aufl.) [Einführung in Automatentheorie, formale Sprachen und Berechenbarkeit](#)
- ▶ Kozen, Dexter C. (1997) [Automata and computability](#)
- ▶ Sipser, Michael (2006, 2. Aufl.) [Introduction to the theory of computation](#)

Diese und noch weitere Bücher zum Thema sind zu finden in der Informatikbibliothek, im Handapparat unter FoSAP.

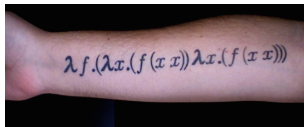
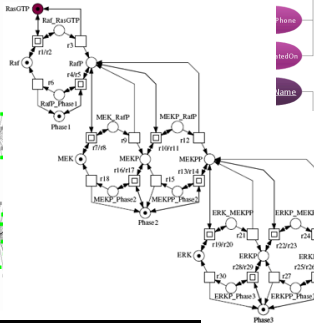
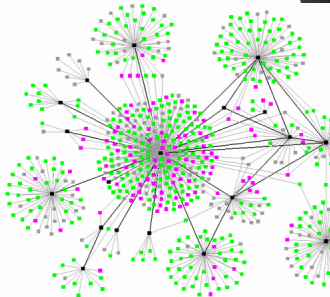
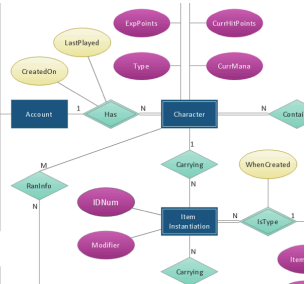
# Einleitung

# Formale Systeme in der Informatik



Japanese Grammar	English Grammar
文 → 適用修飾句 + 文	SENTENCE → SDEC
文 → 用言句	SDEC → SUBJ + VP
適用修飾句 → 用言句 + 適用助詞	SUBJ → NP
適用修飾句 → 名詞句 + 適用助詞	VP → VP + PP
用言句 → 用言句 + 助動詞	SREL → *RELPRO + AUXD + VP
用言句 → 用言	PP → *PREP + NP
用言 → 語幹 + 語尾	VP → *V
名詞句 → 文 + 名詞句	NP → NP + SREL
名詞句 → 名詞	NP → *DET + *N
	AUXD → AUX

ord  
edOn  
Name  
Address  
Email  
Phone  
stedOn  
Name



```

#define PARAMETER_1 0.7
#define PARAMETER_4 0.8

typedef struct RTNode_ {
    int *block;
    int blocksize;
    struct RTNode_ *parent;
} RTNode;

RTNode* gintree(int n) {
    int i;
    RTNode* root;

    root=(RTNode*) calloc(n,sizeof(RTNode));
    if (root != NULL) {
        root->parent = NULL;
        for (i=1; i<n; ++i)
            (root+i)->parent = root + random()%i;
    }
    return(root);
}

int genblocks(RTNode* root, int n, int tw) {
    int i,new,old,bs;
    int *block;
    RTNode *t = root->parent;

```

# Formale Systeme in der Informatik (Forts.)

**Formale Systeme** spielen in der Informatik eine zentrale und vielfältige Rolle,

# Formale Systeme in der Informatik (Forts.)

**Formale Systeme** spielen in der Informatik eine zentrale und vielfältige Rolle, beispielsweise

- ▶ als Programmiersprachen wie **Python**,

# Formale Systeme in der Informatik (Forts.)

**Formale Systeme** spielen in der Informatik eine zentrale und vielfältige Rolle, beispielsweise

- ▶ als Programmiersprachen wie **Python**,
- ▶ als Spezifikationsformalismen wie **UML**,

# Formale Systeme in der Informatik (Forts.)

**Formale Systeme** spielen in der Informatik eine zentrale und vielfältige Rolle, beispielsweise

- ▶ als Programmiersprachen wie **Python**,
- ▶ als Spezifikationsformalismen wie **UML**,
- ▶ zur Beschreibung von Daten und deren Beziehungen untereinander, etwa im **relationalen Datenmodell**,

# Formale Systeme in der Informatik (Forts.)

**Formale Systeme** spielen in der Informatik eine zentrale und vielfältige Rolle, beispielsweise

- ▶ als Programmiersprachen wie **Python**,
- ▶ als Spezifikationsformalismen wie **UML**,
- ▶ zur Beschreibung von Daten und deren Beziehungen untereinander, etwa im **relationalen Datenmodell**,
- ▶ zur Beschreibung von Netzwerken von lokalen Rechnernetzen bis zum WWW und sozialen Netzen, etwa durch **Graphen**,



# Formale Systeme in der Informatik (Forts.)

**Formale Systeme** spielen in der Informatik eine zentrale und vielfältige Rolle, beispielsweise

- ▶ als Programmiersprachen wie **Python**,
- ▶ als Spezifikationsformalismen wie **UML**,
- ▶ zur Beschreibung von Daten und deren Beziehungen untereinander, etwa im **relationalen Datenmodell**,
- ▶ zur Beschreibung von Netzwerken von lokalen Rechnernetzen bis zum WWW und sozialen Netzen, etwa durch **Graphen**,
- ▶ zur Modellierung von Rechnern und Berechnungen, etwa durch **Turing Maschinen**,

# Formale Systeme in der Informatik (Forts.)

**Formale Systeme** spielen in der Informatik eine zentrale und vielfältige Rolle, beispielsweise

- ▶ als Programmiersprachen wie **Python**,
- ▶ als Spezifikationsformalismen wie **UML**,
- ▶ zur Beschreibung von Daten und deren Beziehungen untereinander, etwa im **relationalen Datenmodell**,
- ▶ zur Beschreibung von Netzwerken von lokalen Rechnernetzen bis zum WWW und sozialen Netzen, etwa durch **Graphen**,
- ▶ zur Modellierung von Rechnern und Berechnungen, etwa durch **Turing Maschinen**,
- ▶ zur Modellierung von komplexen Systemen und Prozessen, wie beispielsweise nebenläufigen, verteilten oder mobilen Prozessen, etwa durch **Prozesskalküle**.

# Formale Systeme in der Informatik (Forts.)

Formale Systeme finden man

- ▶ in fast allen Bereichen der Informatik

# Formale Systeme in der Informatik (Forts.)

Formale Systeme finden man

- ▶ in fast allen Bereichen der Informatik
- ▶ auf ganz unterschiedlichen Abstraktionsebenen, von abstrakten mathematischen Kalkülen wie dem  $\lambda$ -Kalkül bis hin zu Assemblersprachen.

Thema dieser Vorlesung sind die Grundlagen der Arbeit mit formalen Systemen.

Thema dieser Vorlesung sind die Grundlagen der Arbeit mit formalen Systemen.

## Zentrale Fragen

- ▶ Wie kann man ein formales System spezifizieren? (Syntax)

Thema dieser Vorlesung sind die Grundlagen der Arbeit mit formalen Systemen.

## Zentrale Fragen

- ▶ Wie kann man ein formales System spezifizieren? (Syntax)
- ▶ Wie kann man den formalen Objekten eine klar definierte Bedeutung geben? (Semantik)

Thema dieser Vorlesung sind die **Grundlagen der Arbeit mit formalen Systemen**.

## Zentrale Fragen

- ▶ Wie kann man ein formales System spezifizieren? (**Syntax**)
- ▶ Wie kann man den formalen Objekten eine klar definierte Bedeutung geben? (**Semantik**)
- ▶ Wie kann man die Eigenschaften eines formalen Systems verstehen?



Thema dieser Vorlesung sind die **Grundlagen der Arbeit mit formalen Systemen**.

## Zentrale Fragen

- ▶ Wie kann man ein formales System spezifizieren? (**Syntax**)
- ▶ Wie kann man den formalen Objekten eine klar definierte Bedeutung geben? (**Semantik**)
- ▶ Wie kann man die Eigenschaften eines formalen Systems verstehen?
- ▶ Wie kann man (auch algorithmisch) mit einem formalen System arbeiten?

## Diese Vorlesung (Forts.)

Wir werden die Antworten auf diese Fragen anhand konkreter formaler Systeme erlernen.

# Diese Vorlesung (Forts.)

Wir werden die Antworten auf diese Fragen anhand konkreter formaler Systeme erlernen.

## Automaten

Endliche Automaten und Verallgemeinerungen von Kellerautomaten bis hin zu Turingmaschinen sind grundlegende Berechnungsmodelle.

# Diese Vorlesung (Forts.)

Wir werden die Antworten auf diese Fragen anhand konkreter formaler Systeme erlernen.

## Automaten

Endliche Automaten und Verallgemeinerungen von Kellerautomaten bis hin zu Turingmaschinen sind grundlegende Berechnungsmodelle.

## Grammatiken

Reguläre Ausdrücke und kontextfreie Grammatiken und allgemeinere Grammatiken sind grundlegende Formalismen zur Spezifikation formaler Sprachen.

Wir werden die Antworten auf diese Fragen anhand konkreter formaler Systeme erlernen.

## Automaten

Endliche Automaten und Verallgemeinerungen von Kellerautomaten bis hin zu Turingmaschinen sind grundlegende Berechnungsmodelle.

## Grammatiken

Reguläre Ausdrücke und kontextfreie Grammatiken und allgemeinere Grammatiken sind grundlegende Formalismen zur Spezifikation formaler Sprachen.

## Nebenläufige Systeme

Die Untersuchung nebenläufiger Systeme stellt eine besondere Herausforderung dar. Petrinetze und Prozesskalküle sind Formalismen zur Spezifikation solcher Systeme.

Bei der Untersuchung der formalen Systeme bedienen wir uns **mathematische Methoden**:

- ▶ exakte Definitionen,
- ▶ formale Beweise,
- ▶ Verwendung mathematischer Strukturen wie Graphen.

Bei der Untersuchung der formalen Systeme bedienen wir uns **mathematische Methoden**:

- ▶ exakte Definitionen,
- ▶ formale Beweise,
- ▶ Verwendung mathematischer Strukturen wie Graphen.

**Algorithmische Fragestellungen** spielen eine gewisse Rolle, stehen aber in dieser Vorlesung nicht im Mittelpunkt.

# Stellung der Vorlesung im Informatikstudium

Teil einer Einführung in die **Theoretische Informatik**



# Stellung der Vorlesung im Informatikstudium

Teil einer Einführung in die **Theoretische Informatik**

## Zwei Teile

**FoSAP** Einführung in Modelle und Sachverhalte, die für die Konstruktion und Analyse von Informatik-Systemen nötig sind.

**BuK** Einführung in die „Naturgesetze der Informationsverarbeitung“  
(z.B. die prinzipiellen Grenzen für den Einsatz von Algorithmen)