

# Informatik IV

## Theoretische Informatik

Formale Sprachen und Automaten,  
Berechenbarkeit und NP-Vollständigkeit

Sommersemester 2019

**Dozent: Prof. Dr. J. Rothe**

im Wechsel mit

Jun.-Prof. Dr. D. Baumeister

Prof. Dr. M. Leuschel



HEINRICH HEINE  
UNIVERSITÄT DÜSSELDORF

# Organisatorisches

- Terminänderungen und aktuelle Meldungen: Vorlesungswebsite
- Studierendenforum im ILIAS:  
Plattform zum Austausch und zur Hilfe untereinander
- Bitte melden Sie sich im LSF für die Vorlesung an!
- Übungen und Lerngruppen: keine Anmeldung erforderlich
- Unterstützungstutorium für die mathematischen Grundlagen zu Informatik I bis IV (von Cynthia Maushagen, siehe LSF und ILIAS)

# Webseiten

- Vorlesungswebseite:

[ccc.cs.uni-duesseldorf.de/~rothe/info4](http://ccc.cs.uni-duesseldorf.de/~rothe/info4)

- **Anmeldung im LSF:**

[lsf.uni-duesseldorf.de](http://lsf.uni-duesseldorf.de)

- Mentoringprogramm:

[www.cs.hhu.de/studium-lehre-informatik/studierende/mentoringprogramm](http://www.cs.hhu.de/studium-lehre-informatik/studierende/mentoringprogramm)

# Übungen (Marc Neveling)

- **Termine:**

- Dienstag 14:30 bis 16:00 im Hörsaal 6C und
- Mittwoch 14:30 bis 16:00 im Hörsaal 6J

- **Übungsblätter:**

- jeden Dienstag auf der Vorlesungswebsite
- Abgabe eine Woche später am Dienstag bis 10:30 Uhr (Briefkästen in 25.13.02)
- Korrektur eine Woche später, Besprechung in den Übungen
- korrigierte Übungsblätter: in den Übungen abholen; sonst in der Sprechstunde von Marc Neveling (25.02-01.35) Mittwoch 11–12
- Abgaben: bitte nicht mit Bleistift schreiben!
- Bepunktungssystem:
  - „X“: Lösung ist falsch oder enthält große Fehler
  - „(OK)“: Lösung enthält kleinere Fehler
  - „OK“: Lösung ist (komplett) richtig

# Lerngruppen

- **Termine:** neun (siehe Vorlesungswebsite)
- **Betreuung:**
  - Anna Kerkmann,
  - Mareike Mutz,
  - Linus Boes und
  - Daniel Neugebauer sowie
  - 10 Tutor/inn/en und Korrektor/inn/en
- **Inhalt:**
  - inhaltliche Fragen zur Vorlesung und zu den Übungsaufgaben
  - gemeinsam zusätzliche Aufgaben mit Unterstützung bearbeiten
  - Wiederholung von Definitionen, Sätzen oder Algorithmen o.Ä. aus der Vorlesung

# Prüfung

- Es gibt **keine Zulassungsbeschränkung** zur Klausur:  
Wer sich fristgerecht zur Klausur anmeldet, darf mitschreiben.
- **WARNUNG:** Trotzdem regelmäßig die Übungsblätter bearbeiten, die Übungen besuchen und bei den Lerngruppen mitmachen! Andernfalls wird es schwer werden, die Klausur zu bestehen.
- *Wir behalten uns das Recht vor, eine Zulassungsbeschränkung einzuführen, falls das Übungsangebot nicht ausreichend genutzt wird. Dies wird natürlich vorher angekündigt.*
- In der Klausur darf alles **Nicht-Elektronische** verwendet werden (Bücher, Skript, Übungsblätter, . . .)
- Es wird im Laufe des Semesters eine **Probeklausur** stattfinden.
- **Klausur: 15.07.2019**, ab 8:30 Uhr (Termine für Klausureinsicht, Nachklausur und Nachklausureinsicht: später)

# Fragen?



# Fragen?

- Lerngruppen
- Übungen
- Vorlesung
- Studierendenforum in ILIAS (ABER: nicht für Lösungsaustausch!)
- Unterstützungstutorium für die mathematischen Grundlagen
- nicht-inhaltliche Fragen und Probleme: Mentoringprogramm
- Mitarbeiter/inn/en:
  - Marc Neveling
  - Anna Kerkmann,
  - Mareike Mutz,
  - Linus Boes und
  - Daniel Neugebauer

## Eigeninitiative!

# Literatur

- Alexander Asteroth und Christel Baier: „Theoretische Informatik. Eine Einführung in Berechenbarkeit, Komplexität und formale Sprachen mit 101 Beispielen“. Pearson Studium, 2002.
- Norbert Blum: „Theoretische Informatik. Eine anwendungsorientierte Einführung“. Oldenbourg, 2. Auflage, 2001.
- Katrin Erk und Lutz Priese: „Theoretische Informatik. Eine umfassende Einführung“. Springer-Verlag, 3. Auflage, 2008.
- Gerhard Goos: „Vorlesungen über Informatik 3. Berechenbarkeit, formale Sprachen, Spezifikationen“. Springer-Verlag, 1997.
- John E. Hopcroft, Rajeev Motwani und Jeffrey D. Ullman: „Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie“. Pearson Studium, 2. Auflage, 2002.

# Literatur

- Jörg Rothe: „Komplexitätstheorie und Kryptologie. Eine Einführung in Kryptokomplexität“. eXamen.press, Springer-Verlag, Berlin, Heidelberg, 2008. (Nur die Abschnitte 2.2 und 3.5.)
- Arto K. Salomaa: „Formale Sprachen“. Springer-Verlag, 1978.
- Uwe Schöning: „Theoretische Informatik – kurz gefasst“. Spektrum Akademischer Verlag, 5. Auflage, 2008.
- Gottfried Vossen und Kurt-Ulrich Witt: „Grundkurs Theoretische Informatik“. Vieweg Verlag, 4. Auflage, 2006.
- Klaus W. Wagner: „Theoretische Informatik. Eine kompakte Einführung“. Springer-Verlag, 2. Auflage, 2003.
- Ingo Wegener: „Theoretische Informatik. Eine algorithmenorientierte Einführung“. Teubner, 3. Auflage, 2005.

# Syntax und Semantik von Computerprogrammen

```
#!/usr/bin/python
def Fib(x):
    if x<2:
        return 1
    else:
        return Fib(x-1)+Fib(x-2)
i = int(raw_input())
print "Fibonacci",i, "=", Fib(i)
```

⇒ Theorie der formalen Sprachen und Automaten

# Andere Programme/Systeme

- Wie soll ein DHCP Server funktionieren?
- Was ist ein korrekter Ablauf bei einem Zugsteuerungssystem?
- Bei einer Waschmaschine mit Aktionen:  
Open, Close, Lock, Unlock, StartWash, StopWash?
  - OK: Close, Lock, StartWash, StopWash, Unlock, Open
  - **Nicht OK**: Close, StartWash
  - **Nicht OK**: Close, Lock, StartWash, Unlock, Open

⇒ Theorie der formalen Sprachen und Automaten

# Wird mein Programm jemals halten?

- Mit dem richtigen Ergebnis?
- Und kann ich das algorithmisch testen?

⇒ Berechenbarkeitstheorie

# One Million Dollar Question: $P = NP?$



**“I can’t find an efficient algorithm,  
but neither can all these famous people.”**

⇒ NP-Vollständigkeit