

Algorithmen und Datenstrukturen

Wintersemester 2022/23

Organisatorisches

- | | | |
|------------------|--------------------------|--|
| Vorlesung: | Alexander Wolff (M4.1.1) | Die ADS ist da für: |
| Übungsbetreuung: | Felix Klesen (M4.1.2) | – B.Sc. (Nachhaltigkeit u.) Informatik |
| Übungen: | Felix Hauser | – B.Sc. Luft- und Raumfahrtinf. |
| | Martin Hesse | – B.Sc. Games Engineering |
| | Antonio Lauerbach | – B.Sc. Math. mit Anwendungsfach Inf. |
| | Niklas Orf | – Lehramt Informatik |
| | Thanh Mai Pham | Die GADS ist da für: |
| | Linus Pleyer | – B.Sc. MCS |
| | Jan Ruppert | – B.Sc. Wirtschaftsinformatik |
| | Samuel Wolf | – B.Sc. Wirtschaftsmathematik |
| | | – ... |

Algorithmen und Datenstrukturen

Wintersemester 2022/23

Organisatorisches

- | | | |
|------------------|--------------------------|--|
| Vorlesung: | Alexander Wolff (M4.1.1) | Die ADS ist da für: |
| Übungsbetreuung: | Felix Klesen (M4.1.2) | – B.Sc. (Nachhaltigkeit u.) Informatik |
| Übungen: | Felix Hauser | – B.Sc. Luft- und Raumfahrtinf. |
| | Martin Hesse | – B.Sc. Games Engineering |
| | Antonio Lauerbach | – B.Sc. Math. mit Anwendungsfach Inf. |
| | Niklas Orf | – Lehramt Informatik |
| | Thanh Mai Pham | Die GADS ist da für: |
| | Linus Pleyer | – B.Sc. MCS |
| | Jan Ruppert | – B.Sc. Wirtschaftsinformatik |
| | Samuel Wolf | – B.Sc. Wirtschaftsmathematik |
| | | – ... |

a) Effiziente Algorithmen

Lehrstuhl für Informatik I

b) Optimierung

Prof. Marie Schmidt

c) Komplexitätstheorie

Prof. Christian Glaßer

a) Effiziente Algorithmen



Alexander Wolff
Professor



Boris Klemz
PostDoc



T. Hegemann



D. Sieper



Felix Klesen



Johannes Zink

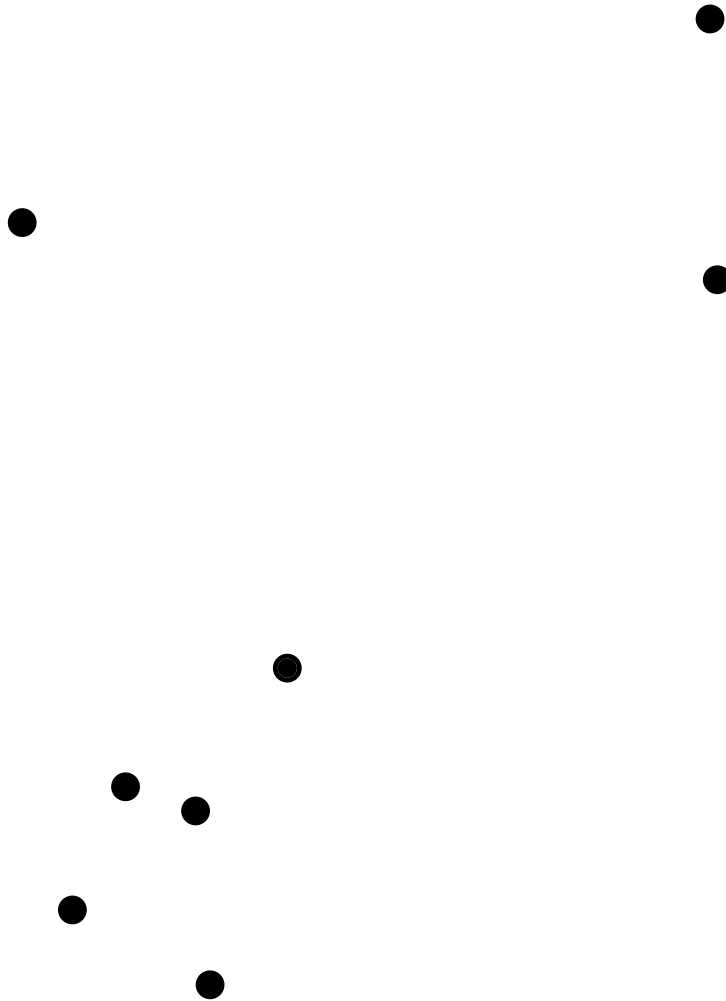


Oksana Firman

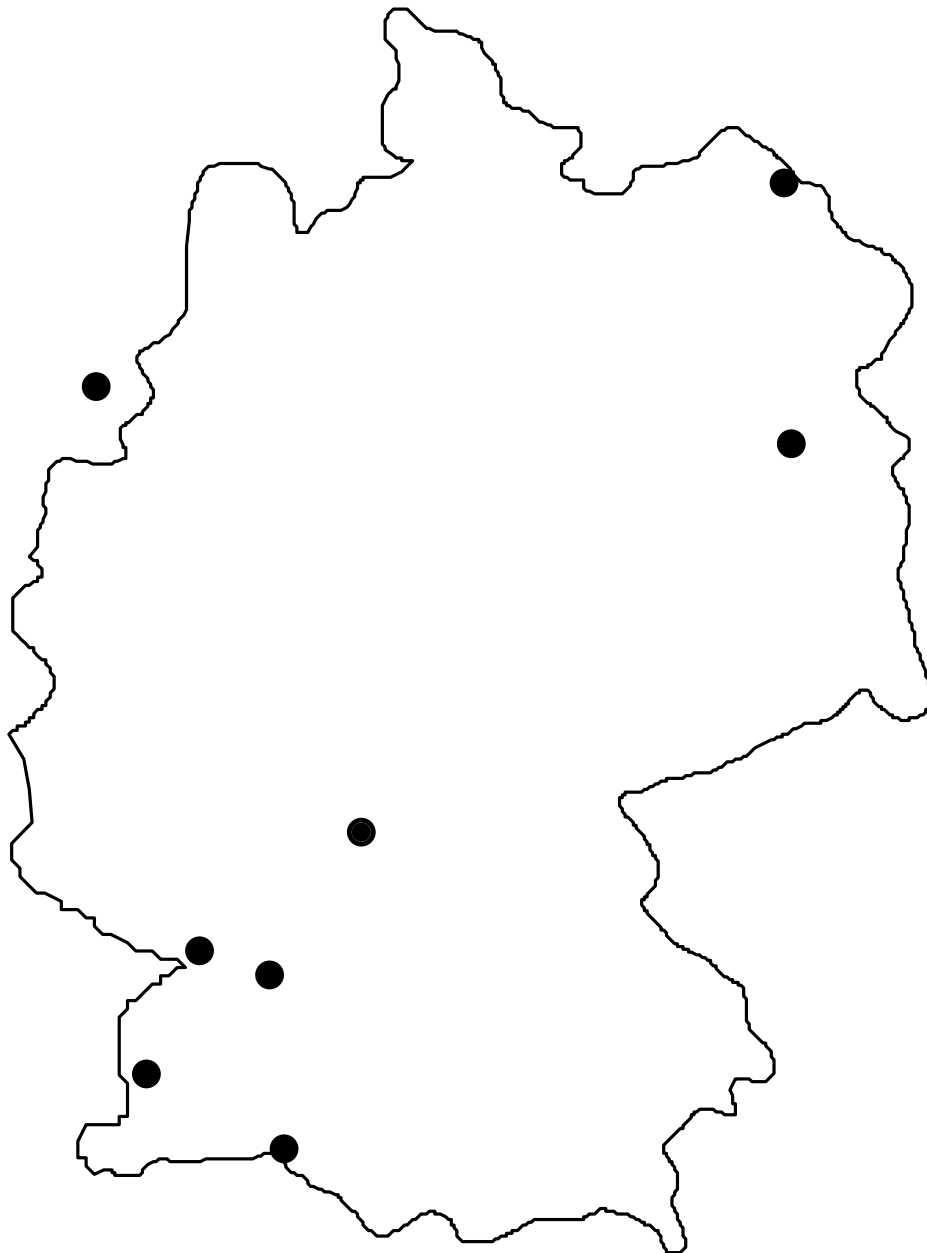


Jakob Geiger

In eigener Sache



In eigener Sache



In eigener Sache



In eigener Sache



In eigener Sache



In eigener Sache



In eigener Sache



In eigener Sache



In eigener Sache



In eigener Sache



In eigener Sache

Alexander Wolff

Email:

`vorname.nachname
@uni-wuerzburg.de`



In eigener Sache



Alexander Wolff

Email:

`vorname.nachname
@uni-wuerzburg.de`

Sprechstunde:

`mittwochs, 13–14h`

In eigener Sache



Alexander Wolff

Email:

`vorname.nachname
@uni-wuerzburg.de`

Sprechstunde:

`mittwochs, 13–14h`

Büro:

`Geb. M4, Raum 1.1`

In eigener Sache



Alexander Wolff

Email:

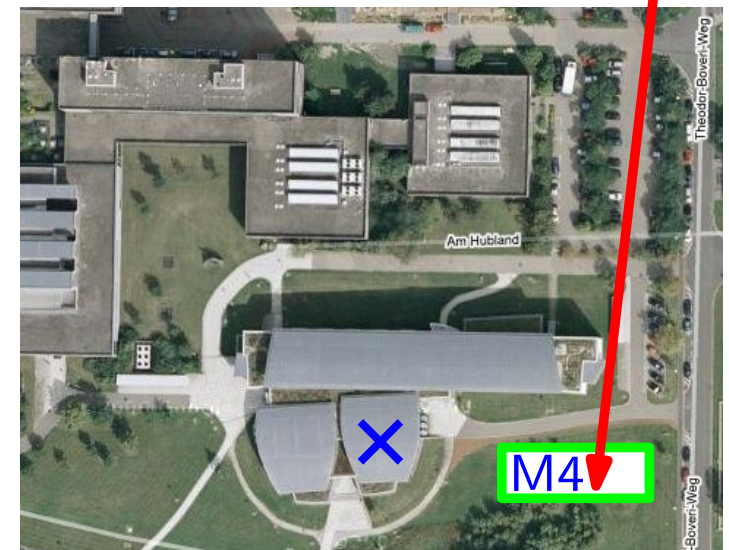
`vorname.nachname
@uni-wuerzburg.de`

Sprechstunde:

mittwochs, 13–14^h

Büro:

Geb. M4, Raum 1.1



In eigener Sache



Felix Klesen,
Raum 1.2,
vorname.nachname@uni-wuerzburg.de

Alexander Wolff

Email:

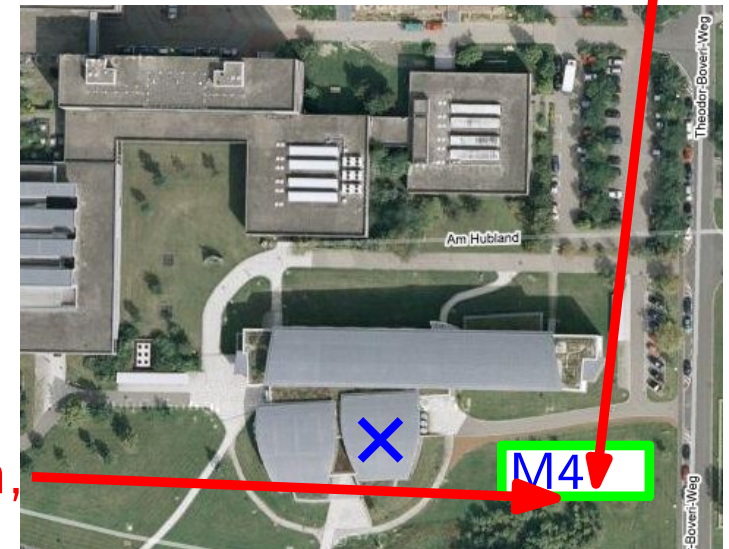
vorname.nachname@uni-wuerzburg.de

Sprechstunde:

mittwochs, 13–14^h

Büro:

Geb. M4, Raum 1.1

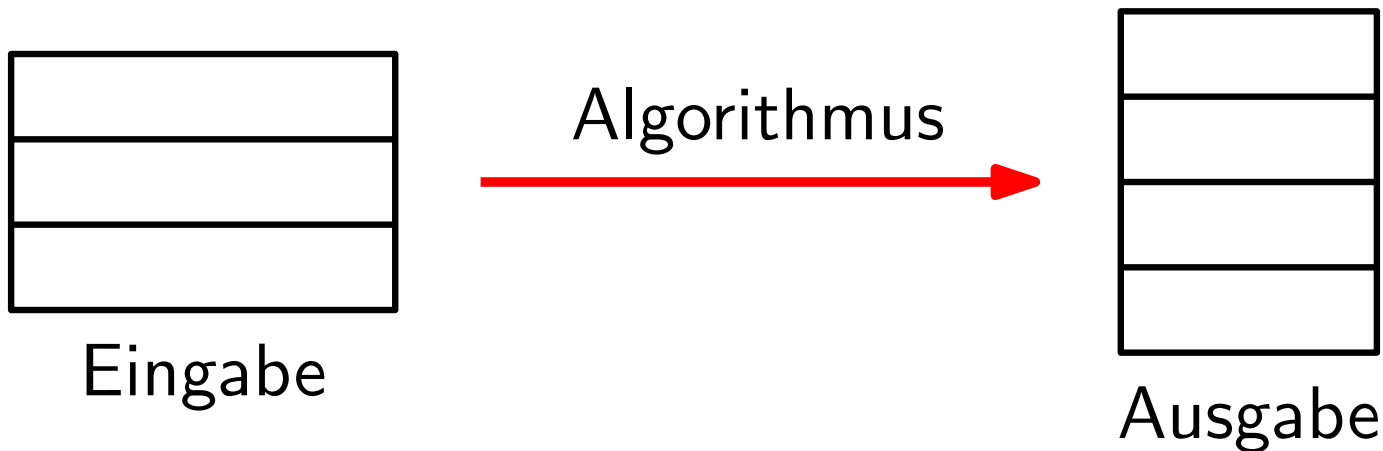


Algorithmen...

... sind (wohldefinierte, endliche) Folgen von Anweisungen, die aus einer Eingabe eine Ausgabe produzieren.

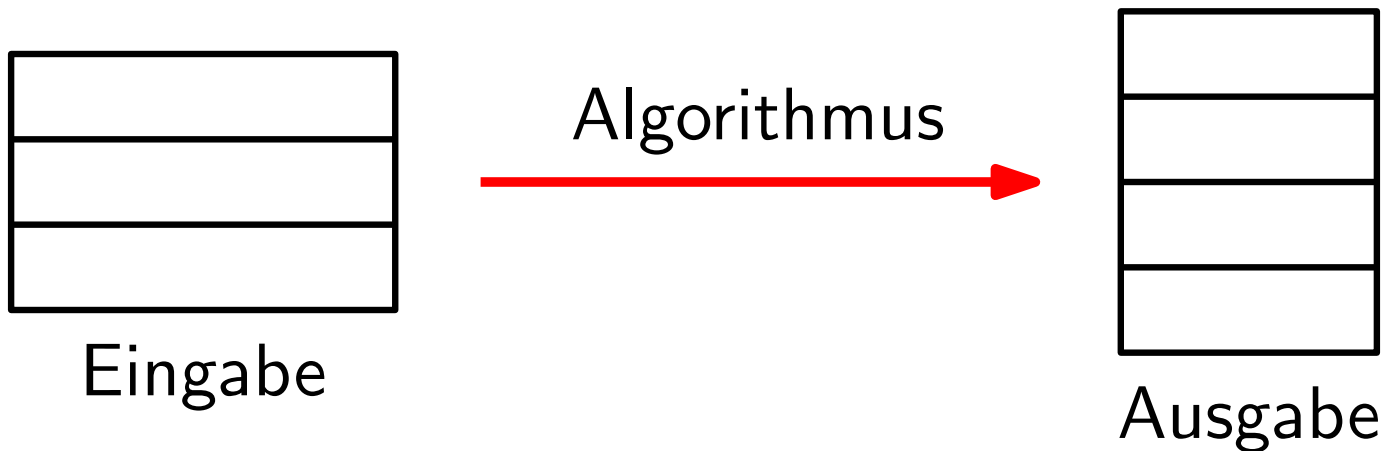
Algorithmen...

... sind (wohldefinierte, endliche) Folgen von Anweisungen, die aus einer Eingabe eine Ausgabe produzieren.



Algorithmen...

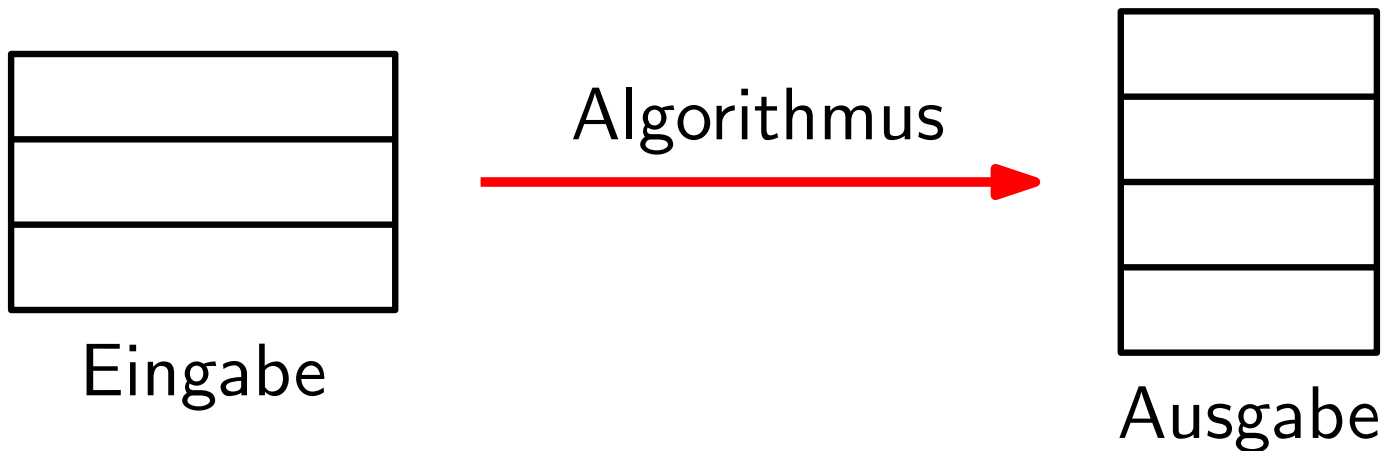
... sind (wohldefinierte, endliche) Folgen von Anweisungen, die aus einer Eingabe eine Ausgabe produzieren.



Beispiele:

Algorithmen...

... sind (wohldefinierte, endliche) Folgen von Anweisungen, die aus einer Eingabe eine Ausgabe produzieren.



Beispiele:

- Kochrezepte
- Algorithmen zur Verknüpfung $(+, -, \cdot, :)$ zweier Zahlen in Dezimaldarstellung
- Euklidscher Algorithmus
- Dijkstras Algorithmus

Algorithmen...

Frage: Ist ein ausführbares Java-Programm ein Algorithmus?

Algorithmen...

Frage: Ist ein ausführbares Java-Programm ein Algorithmus?

„Ein *Algorithmus* ist eine (wohldefinierte, endliche) Folge von Anweisungen, die aus einer Eingabe eine Ausgabe produziert.“

Algorithmen...

Frage: Ist ein ausführbares Java-Programm ein Algorithmus?

„Ein *Algorithmus* ist eine (wohldefinierte, endliche) Folge von Anweisungen, die aus einer Eingabe eine Ausgabe produziert.“

Antwort: • Dem Buchstaben der Definition nach: JA.

Algorithmen...

Frage: Ist ein ausführbares Java-Programm ein Algorithmus?

„Ein *Algorithmus* ist eine (wohldefinierte, endliche) Folge von Anweisungen, die aus einer Eingabe eine Ausgabe produziert.“

Antwort:

- Dem Buchstaben der Definition nach: JA.
- Dem Geiste nach: NEIN. Ich würde sagen:
Ein Algorithmus ist ein abstraktes Konzept;
ein Programm ist eine Instanz dieses Konzeptes.

Algorithmen...

Frage: Ist ein ausführbares Java-Programm ein Algorithmus?

„Ein *Algorithmus* ist eine (wohldefinierte, endliche) Folge von Anweisungen, die aus einer Eingabe eine Ausgabe produziert.“

Antwort:

- Dem Buchstaben der Definition nach: JA.
- Dem Geiste nach: NEIN. Ich würde sagen:
Ein Algorithmus ist ein abstraktes Konzept;
ein Programm ist eine Instanz dieses Konzeptes.



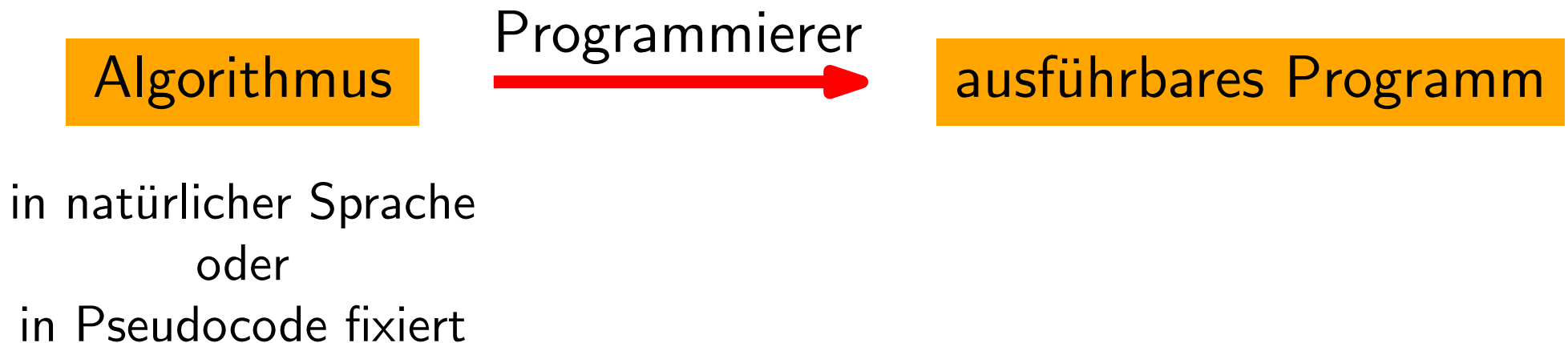
Algorithmen...

Frage: Ist ein ausführbares Java-Programm ein Algorithmus?

„Ein *Algorithmus* ist eine (wohldefinierte, endliche) Folge von Anweisungen, die aus einer Eingabe eine Ausgabe produziert.“

Antwort:

- Dem Buchstaben der Definition nach: JA.
- Dem Geiste nach: NEIN. Ich würde sagen:
Ein Algorithmus ist ein abstraktes Konzept;
ein Programm ist eine Instanz dieses Konzeptes.



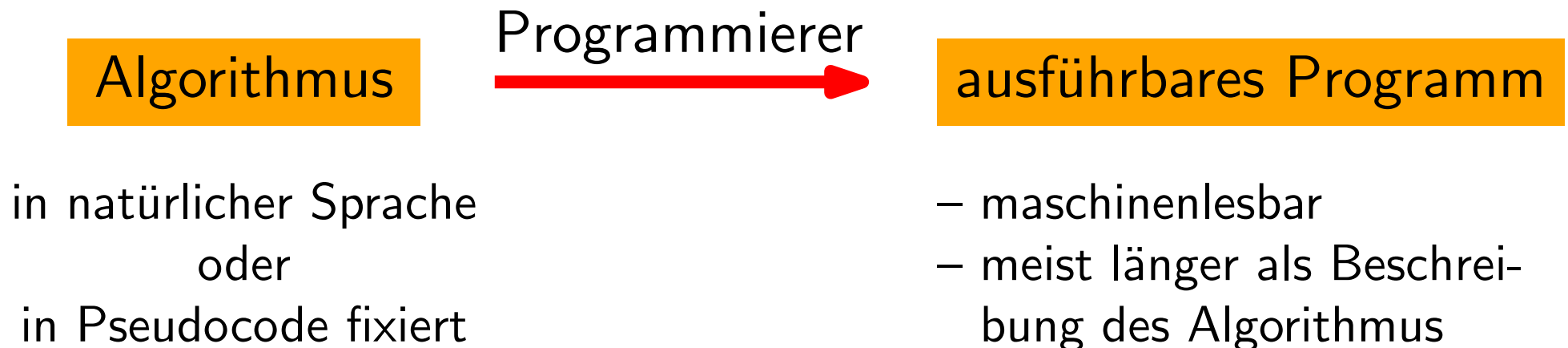
Algorithmen...

Frage: Ist ein ausführbares Java-Programm ein Algorithmus?

„Ein *Algorithmus* ist eine (wohldefinierte, endliche) Folge von Anweisungen, die aus einer Eingabe eine Ausgabe produziert.“

Antwort:

- Dem Buchstaben der Definition nach: JA.
- Dem Geiste nach: NEIN. Ich würde sagen:
Ein Algorithmus ist ein abstraktes Konzept;
ein Programm ist eine Instanz dieses Konzeptes.



... und Datenstrukturen

Datenstruktur:

Konzept, mit dem man Daten speichert und anordnet, so dass man sie schnell finden und ändern kann.

... und Datenstrukturen

Datenstruktur:

Konzept, mit dem man Daten speichert und anordnet, so dass man sie schnell finden und ändern kann.

A large, empty rectangular box with a blue border, likely intended for a diagram or example related to data structures.A second large, empty rectangular box with a blue border, similar to the one above, also likely for a diagram or example.

... und Datenstrukturen

Datenstruktur:

Konzept, mit dem man Daten speichert und anordnet, so dass man sie schnell finden und ändern kann.

Abstrakter Datentyp:

Implementierung:

... und Datenstrukturen

Datenstruktur:

Konzept, mit dem man Daten speichert und anordnet, so dass man sie schnell finden und ändern kann.

Abstrakter Datentyp:

beschreibt die „Schnittstelle“ einer Datenstruktur – welche Operationen werden unterstützt?

Implementierung:

... und Datenstrukturen

Datenstruktur:

Konzept, mit dem man Daten speichert und anordnet, so dass man sie schnell finden und ändern kann.

Abstrakter Datentyp:

beschreibt die „Schnittstelle“ einer Datenstruktur – welche Operationen werden unterstützt?

Implementierung:

wie wird die gewünschte Funktionalität realisiert:

- wie sind die Daten gespeichert (Feld, Liste, ...)?
- welche Algorithmen implementieren die Operationen?

Algorithmen & Datenstrukturen

Lernziele: In dieser Veranstaltung werden Sie lernen...

Algorithmen & Datenstrukturen

Lernziele: In dieser Veranstaltung werden Sie lernen...

- die Effizienz von Algorithmen zu messen und miteinander zu vergleichen,
- grundlegende Algorithmen und Datenstrukturen in Java zu implementieren,
- selbst Algorithmen und Datenstrukturen zu entwerfen sowie
- deren Korrektheit und Effizienz zu beweisen.

Algorithmen & Datenstrukturen

Lernziele: In dieser Veranstaltung werden Sie lernen...

- die Effizienz von Algorithmen zu **messen** und miteinander zu **vergleichen**,
- grundlegende Algorithmen und Datenstrukturen in Java zu implementieren,
- selbst Algorithmen und Datenstrukturen zu entwerfen sowie
- deren Korrektheit und Effizienz zu beweisen.

Algorithmen & Datenstrukturen

Lernziele: In dieser Veranstaltung werden Sie lernen...

- die Effizienz von Algorithmen zu **messen** und miteinander zu **vergleichen**,
- grundlegende Algorithmen und Datenstrukturen in Java zu **implementieren**,
- selbst Algorithmen und Datenstrukturen zu entwerfen sowie
- deren Korrektheit und Effizienz zu beweisen.

Algorithmen & Datenstrukturen

Lernziele: In dieser Veranstaltung werden Sie lernen...

- die Effizienz von Algorithmen zu **messen** und miteinander zu **vergleichen**,
- grundlegende Algorithmen und Datenstrukturen in Java zu **implementieren**,
- selbst Algorithmen und Datenstrukturen zu **entwerfen** sowie
- deren Korrektheit und Effizienz zu beweisen.

Algorithmen & Datenstrukturen

Lernziele: In dieser Veranstaltung werden Sie lernen...

- die Effizienz von Algorithmen zu messen und miteinander zu vergleichen,
- grundlegende Algorithmen und Datenstrukturen in Java zu implementieren,
- selbst Algorithmen und Datenstrukturen zu entwerfen sowie
- deren Korrektheit und Effizienz zu beweisen.

Algorithmen & Datenstrukturen

Lernziele: In dieser Veranstaltung werden Sie lernen...

- die Effizienz von Algorithmen zu **messen** und miteinander zu **vergleichen**,
- grundlegende Algorithmen und Datenstrukturen in Java zu **implementieren**,
- selbst Algorithmen und Datenstrukturen zu **entwerfen** sowie
- deren Korrektheit und Effizienz zu **beweisen**.

Inhalt:

- Grundlagen und Analysetechniken
- Sortierverfahren
- *Entwurfstechniken* für Algorithmen
- Datenstrukturen
- Algorithmen für Graphen
- Systematisches Probieren

Algorithmen & Datenstrukturen

Lernziele: In dieser Veranstaltung werden Sie lernen...

- die Effizienz von Algorithmen zu **messen** und miteinander zu **vergleichen**,
- grundlegende Algorithmen und Datenstrukturen in Java zu **implementieren**,
- selbst Algorithmen und Datenstrukturen zu **entwerfen** sowie
- deren Korrektheit und Effizienz zu **beweisen**.

Inhalt:

- Grundlagen und Analysetechniken
- Sortierverfahren
- **Entwurfstechniken** für Algorithmen
- Datenstrukturen
- Algorithmen für Graphen
- Systematisches Probieren

Ihre Voraussetzungen

- Schulmathematik, insbesondere:

Ihre Voraussetzungen

- Schulmathematik, insbesondere:
 - Grundrechenarten & Logarithmus

Ihre Voraussetzungen

- Schulmathematik, insbesondere:
 - Grundrechenarten & Logarithmus

$$\text{z.B. } \frac{\log_b x}{\log_b y} = ?$$

Ihre Voraussetzungen

- Schulmathematik, insbesondere:
 - Grundrechenarten & Logarithmus
 - Drei Summen:
 - 1) $\sum_{i=1}^n i$
 - 2) $\sum_{i=0}^n q^i$
 - 3) $\sum_{i=1}^n \frac{1}{i}$

z.B. $\frac{\log_b x}{\log_b y} = ?$

Ihre Voraussetzungen

- Schulmathematik, insbesondere:
 - Grundrechenarten & Logarithmus

$$\text{z.B. } \frac{\log_b x}{\log_b y} = ?$$

- Drei Summen:
- 1) $\sum_{i=1}^n i$ *arithmetische Reihe*
 - 2) $\sum_{i=0}^n q^i$
 - 3) $\sum_{i=1}^n \frac{1}{i}$

Ihre Voraussetzungen

- Schulmathematik, insbesondere:

- Grundrechenarten & Logarithmus

$$\text{z.B. } \frac{\log_b x}{\log_b y} = ?$$

- Drei Summen:

1) $\sum_{i=1}^n i$	<i>arithmetische Reihe</i>
2) $\sum_{i=0}^n q^i$	<i>geometrische Reihe</i>
3) $\sum_{i=1}^n \frac{1}{i}$	

Ihre Voraussetzungen

- Schulmathematik, insbesondere:

- Grundrechenarten & Logarithmus

z.B. $\frac{\log_b x}{\log_b y} = ?$

- Drei Summen:

1) $\sum_{i=1}^n i$	<i>arithmetische Reihe</i>
2) $\sum_{i=0}^n q^i$	<i>geometrische Reihe</i>
3) $\sum_{i=1}^n \frac{1}{i}$	<i>harmonische Reihe</i>

Ihre Voraussetzungen

- Schulmathematik, insbesondere:

- Grundrechenarten & Logarithmus

z.B. $\frac{\log_b x}{\log_b y} = ?$

- Drei Summen: 1) $\sum_{i=1}^n i$ *arithmetische Reihe*

- 2) $\sum_{i=0}^n q^i$ *geometrische Reihe*

- 3) $\sum_{i=1}^n \frac{1}{i}$ *harmonische Reihe*

- Linearität des Erwartungswerts

Ihre Voraussetzungen

- Schulmathematik, insbesondere:

- Grundrechenarten & Logarithmus

z.B. $\frac{\log_b x}{\log_b y} = ?$

- Drei Summen: 1) $\sum_{i=1}^n i$ *arithmetische Reihe*

2) $\sum_{i=0}^n q^i$ *geometrische Reihe*

3) $\sum_{i=1}^n \frac{1}{i}$ *harmonische Reihe*

- Linearität des Erwartungswerts

$$E[X + Y] = E[X] + E[Y]$$

Ihre Voraussetzungen

- Schulmathematik, insbesondere:

- Grundrechenarten & Logarithmus

z.B. $\frac{\log_b x}{\log_b y} = ?$

- Drei Summen: 1) $\sum_{i=1}^n i$ *arithmetische Reihe*

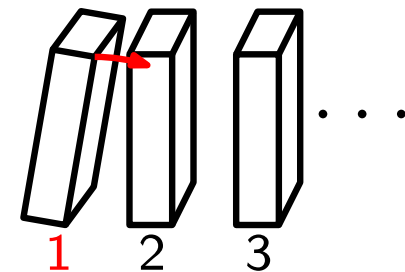
2) $\sum_{i=0}^n q^i$ *geometrische Reihe*

3) $\sum_{i=1}^n \frac{1}{i}$ *harmonische Reihe*

- Linearität des Erwartungswerts

$$E[X + Y] = E[X] + E[Y]$$

- Beweise mit vollständiger Induktion



Ihre Voraussetzungen

- Schulmathematik, insbesondere:

- Grundrechenarten & Logarithmus

z.B. $\frac{\log_b x}{\log_b y} = ?$

- Drei Summen: 1) $\sum_{i=1}^n i$ *arithmetische Reihe*

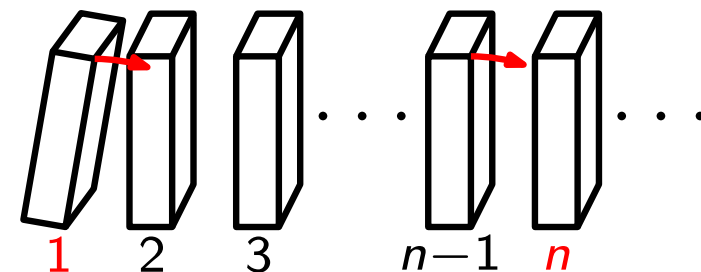
2) $\sum_{i=0}^n q^i$ *geometrische Reihe*

3) $\sum_{i=1}^n \frac{1}{i}$ *harmonische Reihe*

- Linearität des Erwartungswerts

$$E[X + Y] = E[X] + E[Y]$$

- Beweise mit vollständiger Induktion



Ihre Voraussetzungen

- Schulmathematik, insbesondere:

- Grundrechenarten & Logarithmus

z.B. $\frac{\log_b x}{\log_b y} = ?$

- Drei Summen: 1) $\sum_{i=1}^n i$ *arithmetische Reihe*

2) $\sum_{i=0}^n q^i$ *geometrische Reihe*

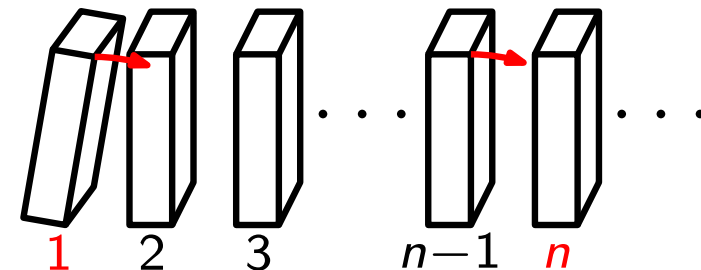
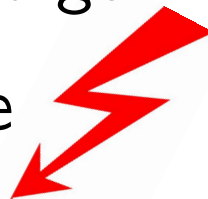
3) $\sum_{i=1}^n \frac{1}{i}$ *harmonische Reihe*

- Linearität des Erwartungswerts

$$E[X + Y] = E[X] + E[Y]$$

- Beweise mit vollständiger Induktion

- Widerspruchsbeweise



Ihre Voraussetzungen

- Schulmathematik, insbesondere:

- Grundrechenarten & Logarithmus

z.B. $\frac{\log_b x}{\log_b y} = ?$

- Drei Summen: 1) $\sum_{i=1}^n i$ *arithmetische Reihe*

2) $\sum_{i=0}^n q^i$ *geometrische Reihe*

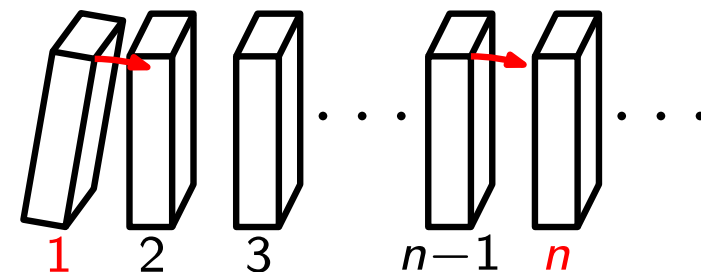
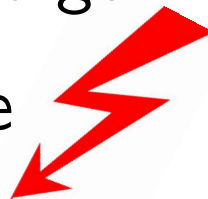
3) $\sum_{i=1}^n \frac{1}{i}$ *harmonische Reihe*

- Linearität des Erwartungswerts

$$E[X + Y] = E[X] + E[Y]$$

- Beweise mit vollständiger Induktion

- Widerspruchsbeweise



- Bereitschaft sich in Java hineinzudenken und -zuüben



Ihre Voraussetzungen

- Schulmathematik, insbesondere:

- Grundrechenarten & Logarithmus

z.B. $\frac{\log_b x}{\log_b y} = ?$

- Drei Summen: 1) $\sum_{i=1}^n i$ *arithmetische Reihe*

2) $\sum_{i=0}^n q^i$ *geometrische Reihe*

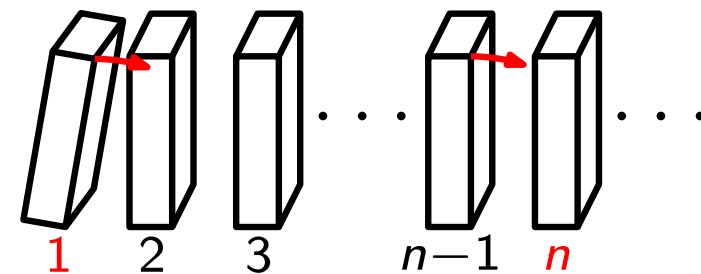
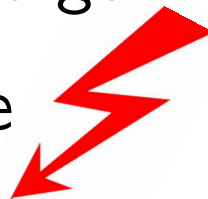
3) $\sum_{i=1}^n \frac{1}{i}$ *harmonische Reihe*

- Linearität des Erwartungswerts

$$E[X + Y] = E[X] + E[Y]$$

- Beweise mit vollständiger Induktion

- Widerspruchsbeweise



- Bereitschaft sich in Java hineinzudenken und -zuüben

- Keine Angst vorm Fragenstellen!!!



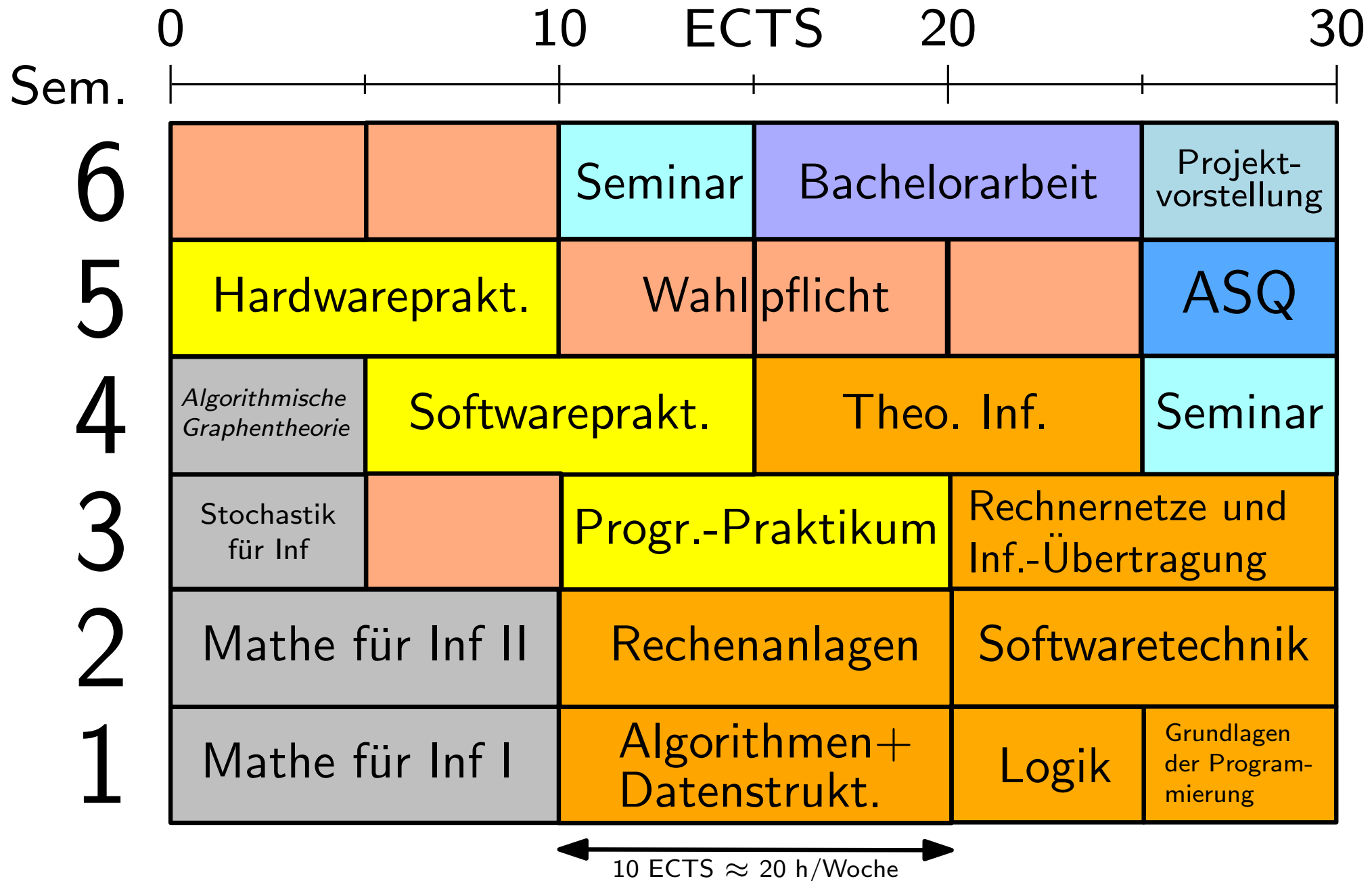
Studienverlaufsplan BA Informatik (Start WS)

SWS: ca. 20

Sem.	ECTS				
	0	10	20	30	
6			Seminar	Bachelorarbeit	Projekt- vorstellung
5	Hardwareprakt.		Wahlpflicht		ASQ
4	<i>Algorithmische Graphentheorie</i>	Softwareprakt.		Theo. Inf.	Seminar
3	Stochastik für Inf		Progr.-Praktikum		Rechnernetze und Inf.-Übertragung
2	Mathe für Inf II		Rechenanlagen	Softwaretechnik	
1	Mathe für Inf I		Algorithmen+ Datenstrukt.	Logik	Grundlagen der Program- mierung

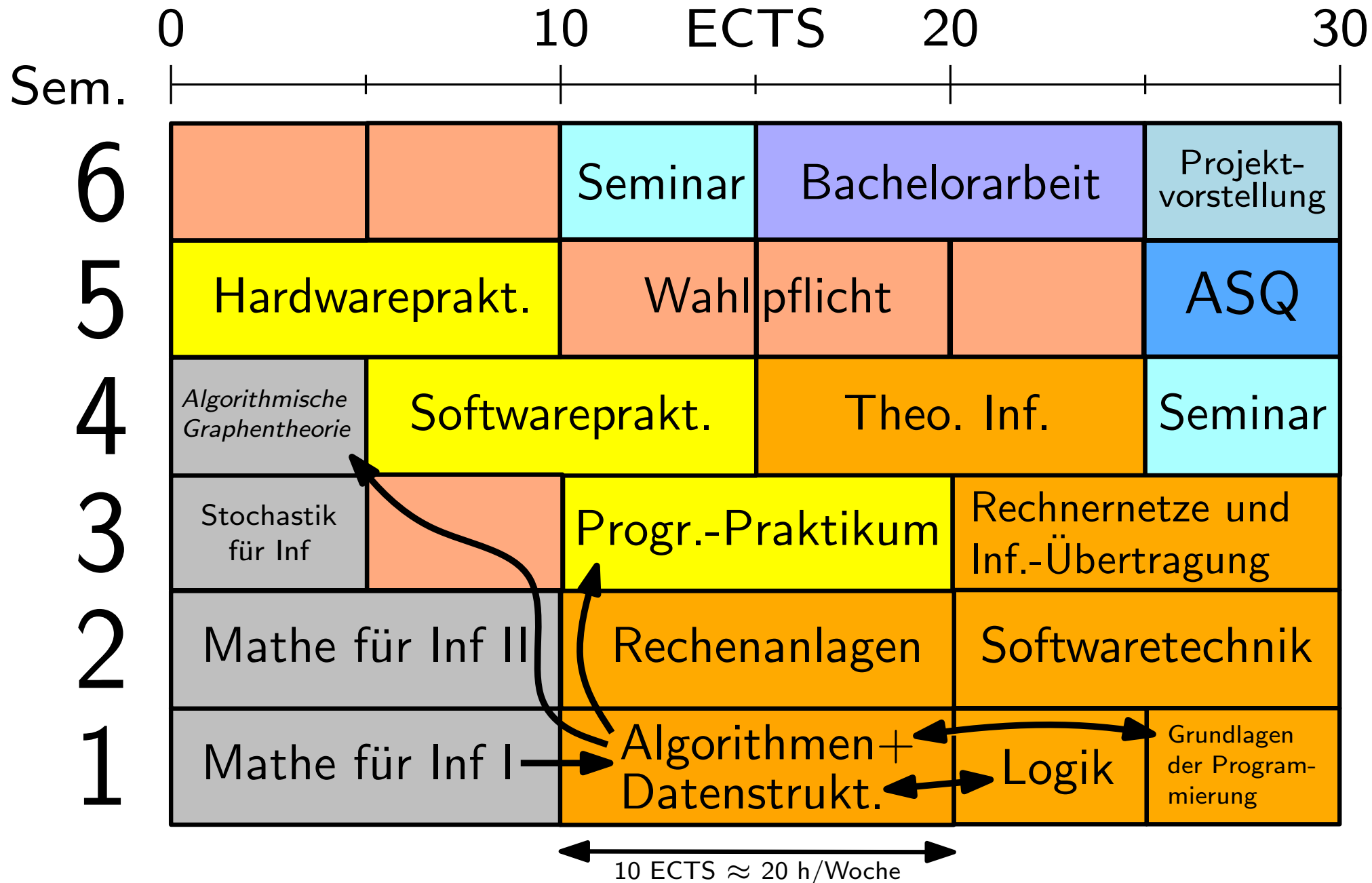
Studienverlaufsplan BA Informatik (Start WS)

SWS: ca. 20



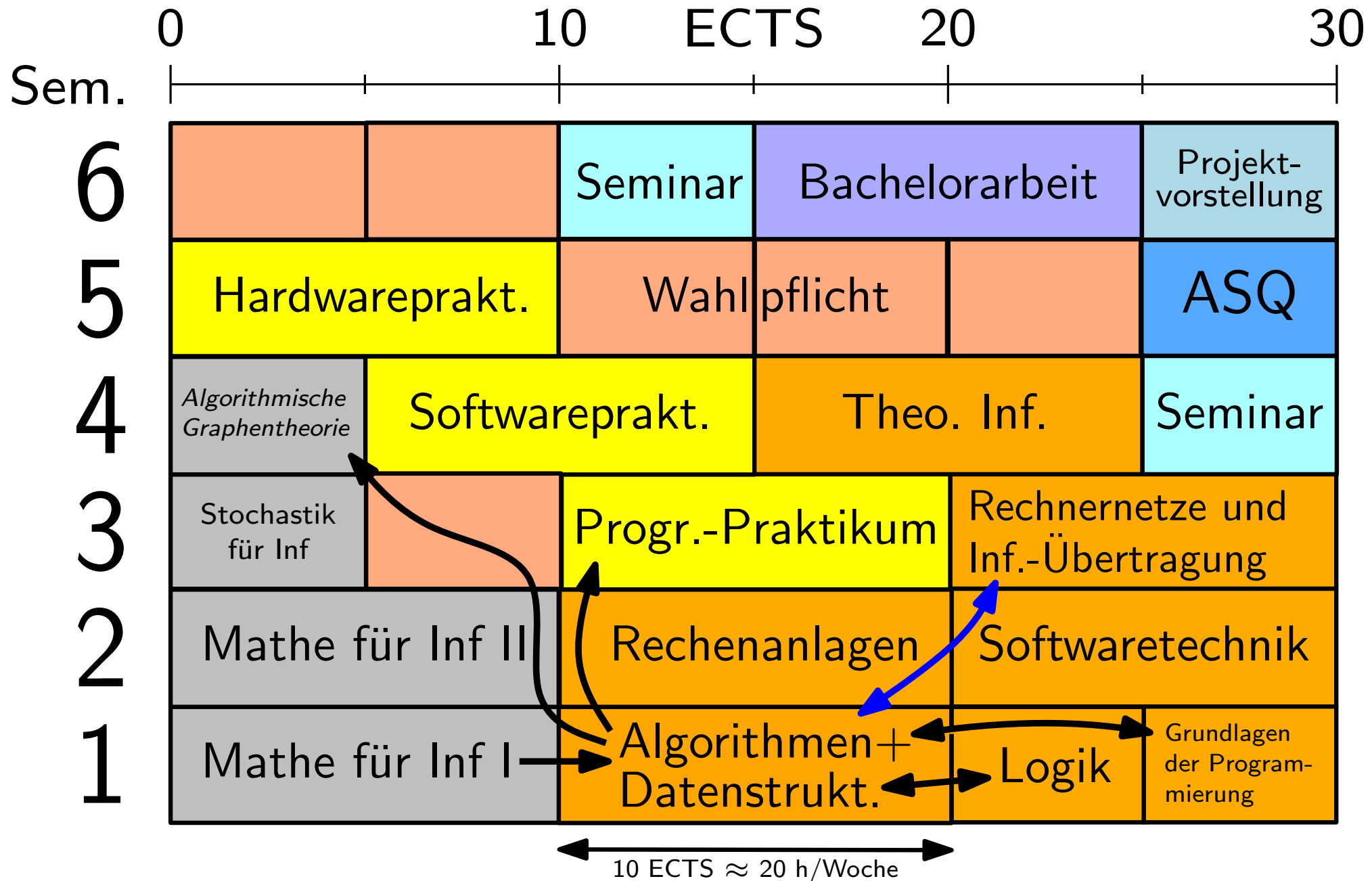
Studienverlaufsplan BA Informatik (Start WS)

SWS: ca. 20



Studienverlaufsplan BA Informatik (Start WS)

SWS: ca. 20



Organisation I: Wochenplan

	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
8 ²⁵ –10				Vorlesung <i>Zuse-HS</i>	
10–12		Vorlesung <i>Zuse-HS</i>			Ausgabe Ü-Blätter <i>WueCampus</i>
12–14		Übung <i>ÜRI & SE 8</i>	Übung <i>SE II</i>	Abgabe Lösungen 14:00	
14–16		Übung <i>ÜR II & SE I</i>	Übung <i>SE I</i>		
16–18		Übung <i>ÜR II</i>	Übung <i>SE II</i>		

Organisation I: Wochenplan

	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
8 ²⁵ –10				Vorlesung <i>Zuse-HS</i>	
10–12		Vorlesung <i>Zuse-HS</i>			Ausgabe Ü-Blätter <i>WueCampus</i>
12–14		Übung <i>ÜR I & SE 8</i>	Übung <i>SE II</i>	Abgabe Lösungen 14:00	
14–16		Übung <i>ÜR II & SE I</i>	Übung <i>SE I</i>		1. <i>WueCampus</i> 2. <i>PABS</i>
16–18		Übung <i>ÜR II</i>	Übung <i>SE II</i>		

Organisation I: Wochenplan

	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
8 ²⁵ –10				Vorlesung <i>Zuse-HS</i>	
10–12		Vorlesung <i>Zuse-HS</i>			Ausgabe Ü-Blätter <i>WueCampus</i>
12–14	0. Übungsblatt schon da + freiwillig	Übung <i>ÜR I & SE 8</i>	Übung <i>SE II</i>	Abgabe Lösungen 14:00	
14–16	Finden diese Woche (fast) alle statt!	Übung <i>ÜR II & SE I</i>	Übung <i>SE I</i>		1. <i>WueCampus</i> 2. <i>PABS</i>
16–18	Diese Woche (fast) beliebig aussuchen!	Übung <i>ÜR II</i>	Übung <i>SE II</i>		

Organisation I: Wochenplan

	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
8 ²⁵ –10				Vorlesung <i>Zuse-HS</i>	
10–12		Vorlesung <i>Zuse-HS</i>			Ausgabe Ü-Blätter <i>WueCampus</i>
12–14	0. Übungsblatt schon da + freiwillig	Übung <i>ÜR I & SE 8</i>	Übung <i>SE II</i>	Abgabe Lösungen 14:00	
14–16	Finden diese Woche (fast) alle statt!	Übung <i>ÜR II & SE I</i>	Übung <i>SE I</i>		1. WueCampus 2. PABS
16–18	Diese Woche (fast) beliebig aussuchen!	Übung <i>ÜR II</i>	Übung <i>SE II</i>	Gruppeneinteilung bis heute, 18.10., 23:59 . Geben Sie drei Termine an! Das erhöht die Wahrscheinlichkeit, einen davon zu bekommen.	

Organisation I: Wochenplan

	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
8 ²⁵ –10				Vorlesung <i>Zuse-HS</i>	
10–12	0. Übungsblatt schon da + freiwillig	Vorlesung <i>Zuse-HS</i>	Ab und zu Laptop nötig!		Ausgabe Ü-Blätter <i>WueCampus</i>
12–14	Finden diese Woche (fast) alle statt!	Übung <i>ÜR I & SE 8</i>	Übung <i>SE II</i>	Abgabe Lösungen 14:00	
14–16		Übung <i>ÜR II & SE I</i>	Übung <i>SE I</i>		1. WueCampus 2. PABS
16–18	Diese Woche (fast) beliebig aussuchen!	Übung <i>ÜR II</i>	Übung <i>SE II</i>	Gruppeneinteilung bis heute, 18.10., 23:59 . Geben Sie drei Termine an! Das erhöht die Wahrscheinlichkeit, einen davon zu bekommen.	

Organisation II: Semesterplan

Di, 18.10. Start Vorlesung & Übungen

Organisation II: Semesterplan

Di, 18.10. Start Vorlesung & Übungen

Do, 17.11. 1. Zwischentest

Organisation II: Semesterplan

Di, 18.10. Start Vorlesung & Übungen

Do, 17.11. 1. Zwischentest

Do, 15.12. 2. Zwischentest

Organisation II: Semesterplan

Di, 18.10. Start Vorlesung & Übungen

Do, 17.11. 1. Zwischentest

Do, 15.12. 2. Zwischentest

24.12.–06.01. Weihnachtsferien

Organisation II: Semesterplan

Di, 18.10. Start Vorlesung & Übungen

Do, 17.11. 1. Zwischentest

Do, 15.12. 2. Zwischentest

24.12.–06.01. Weihnachtsferien

Do, 19.01. 3. Zwischentest

Organisation II: Semesterplan

Di, 18.10. Start Vorlesung & Übungen

Do, 17.11. 1. Zwischentest

Do, 15.12. 2. Zwischentest

24.12.–06.01. Weihnachtsferien

Do, 19.01. 3. Zwischentest

Do, 09.02. Letzte ADS-Vorlesung

Organisation II: Semesterplan

Di, 18.10. Start Vorlesung & Übungen

Do, 17.11. 1. Zwischentest

Do, 15.12. 2. Zwischentest

24.12.–06.01. Weihnachtsferien

Do, 19.01. 3. Zwischentest

Do, 09.02. Letzte ADS-Vorlesung

XX, ??..02. 1. Klausur (AOK-HS u.a.??)

Organisation II: Semesterplan

Di, 18.10. Start Vorlesung & Übungen

Do, 17.11. 1. Zwischentest

Do, 15.12. 2. Zwischentest

24.12.–06.01. Weihnachtsferien

Do, 19.01. 3. Zwischentest

Do, 09.02. Letzte ADS-Vorlesung

XX, ??..02. 1. Klausur (AOK-HS u.a.??)

XX, ??..04. 2. Klausur (Turing-HS, Zuse-HS, HS 2 NW-HSG ?)

Organisatorisches III: Anforderungen ADS

Studienordnung > 2014

1 Modul

- **Übung:**

- **50%** aller Punkte in den Übungen (Arbeit in 2er/3er-Gruppen) und den Zwischentests (Einzelarbeit)
- **0%** Plagiate

- **Vorlesung:**

- Vorlesung + Klausur (benotet)
- Sie dürfen (im Prinzip bel. oft) wiederholen, *solange Sie nicht bestehen.*

Organisatorisches III: Anforderungen ADS

Studienordnung > 2014

1 Modul

- **Übung:**

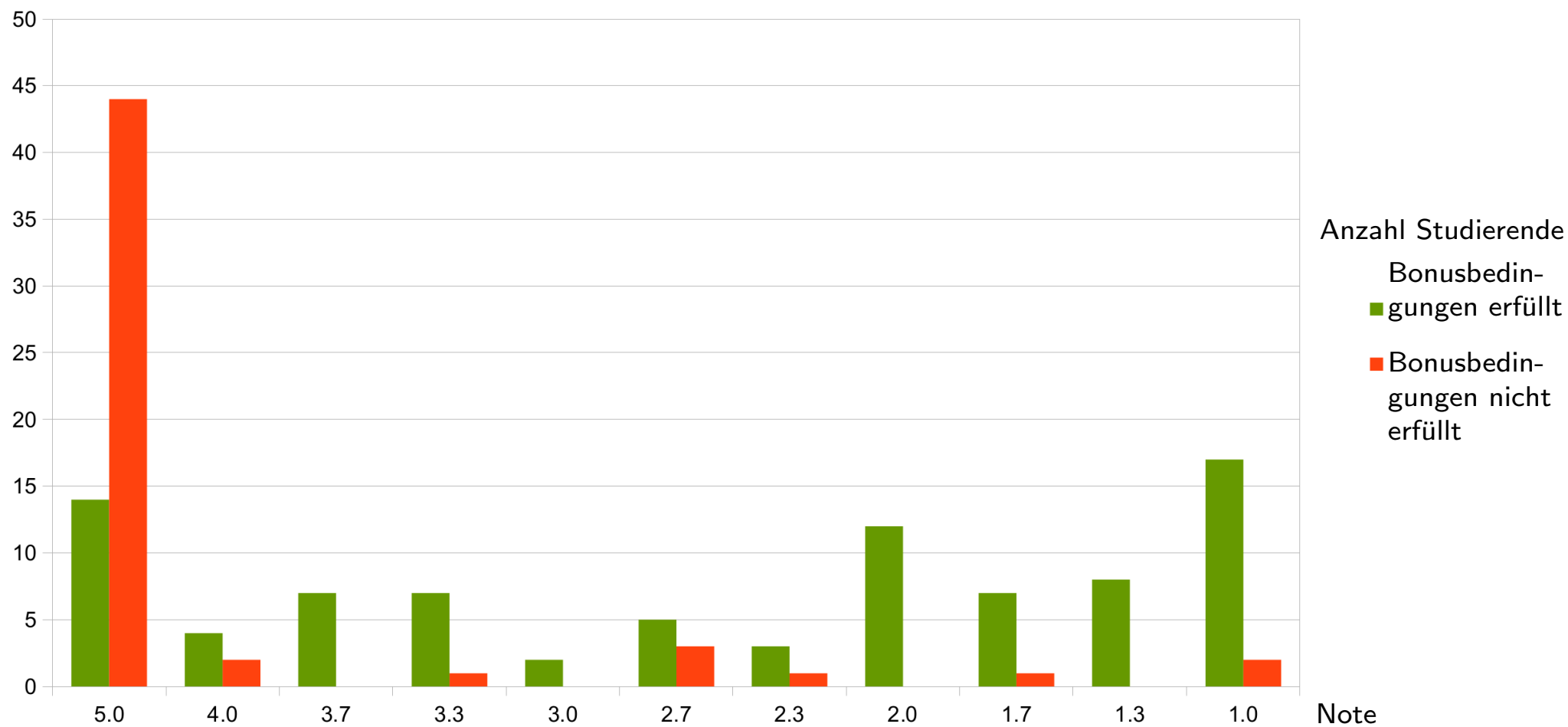
- **50%** aller Punkte in den Übungen (Arbeit in 2er/3er-Gruppen) und den Zwischentests (Einzelarbeit)
- **0%** Plagiate

Das ist die Voraussetzung für den Bonus (0,3 Notenpunkte – bei Bestehen der Klausur)

- **Vorlesung:**

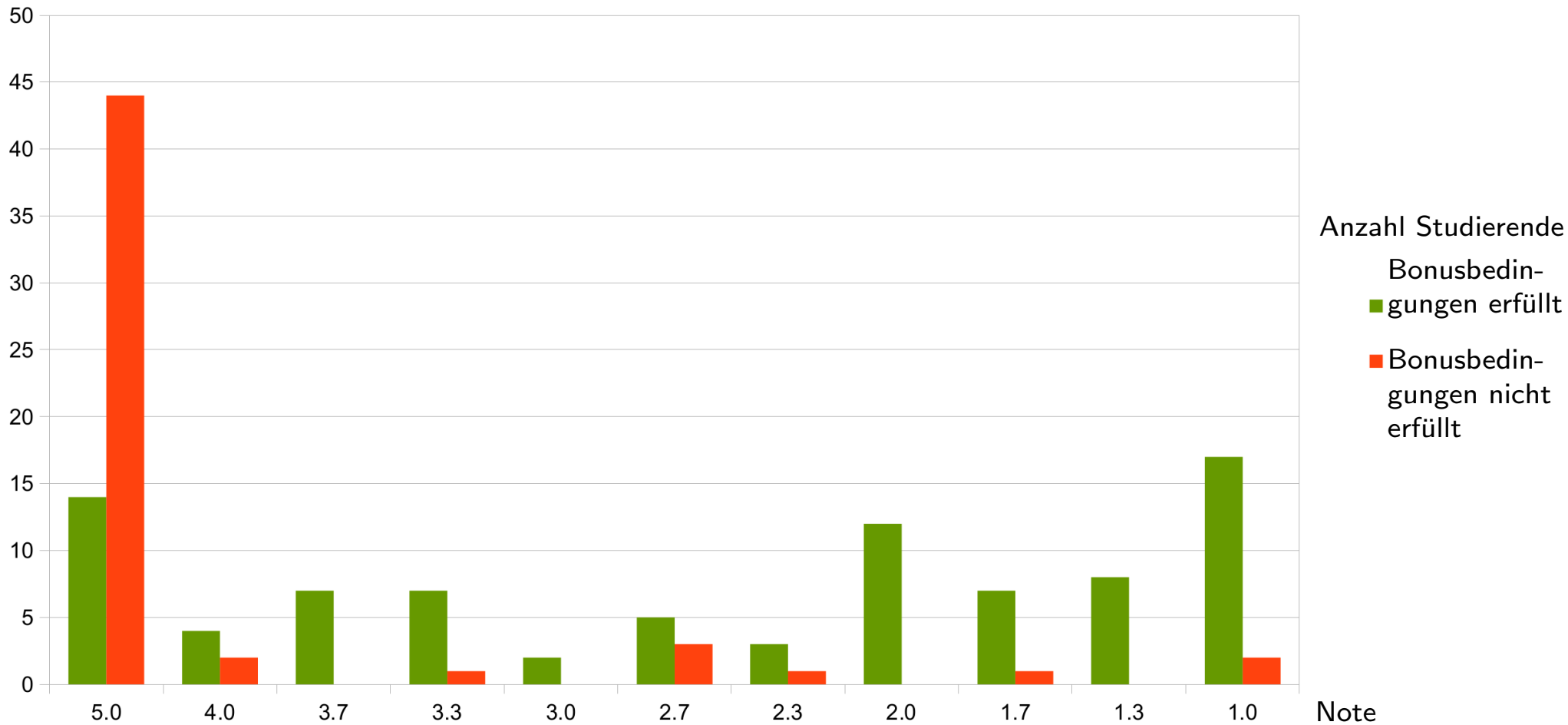
- Vorlesung + Klausur (benotet)
- Sie dürfen (im Prinzip bel. oft) wiederholen, *solange Sie nicht bestehen.*

Motivation Bonus



Klausurergebnisse ADS-Klausur vom 8.2.2016

Motivation Bonus



Klausurergebnisse ADS-Klausur vom 8.2.2016

Anteil „bestanden“ unter denen, die die Bonusbedingungen erfüllt haben: 83,7 %

Anteil „bestanden“ unter denen, die die Bonusbedingungen nicht erfüllt haben: 18,5 %

Wer nicht kommt, verliert

VON JAN-MARTIN WIARDA

Der Jubel war groß, als Universitäten die Anwesenheitspflicht abschafften. Nun zeigt eine Studie: Die Noten werden schlechter.



© David-W - Photocase.de

Wie wirkt sich die Lernbereitschaft auf die Abschaffung der Anwesenheitspflicht aus?

»Anwesenheitsobliegenheiten«. »Wie soll denn das beides zusammengehen?«, fragte sich Schulmeister. »Wie kann man jemandem den Studienerfolg garantieren, der nicht das Studienangebot wahrnimmt?« Der Ehrgeiz des Forschers war geweckt. Jetzt hat Schulmeister eine Metastudie vorgelegt, die 298 Studien zur studentischen Anwesenheit auswertet, aus 25 Ländern und sieben Jahrzehnten. Das Ergebnis: **Es gibt einen klaren Zusammenhang zwischen der Anwesenheit der Studenten in den Lehrveranstaltungen und ihrem Studienerfolg.** Konkret: Je nach Studie reichen schon drei verpasste Termine, um signifikant schlechter in Prüfungen abzuschneiden. Laut mehreren Autoren liegt die Schwelle, ab der die Leistung merklich sinkt, bei vier Abwesenheiten. Werte, die weitgehend unabhängig vom Entstehungsort oder Zeitpunkt der Studie sind.

So weit, so trivial? Die NRW-Ministerin Schulze sagt: »Erst wenn eine Studie zeigen würde, dass man Prüfungen besser bewältigt, wenn man nicht anwesend ist, würde ich ernsthaft anfangen, mir Sorgen um die Hochschulen zu machen.« Ben Seel, Vorstandsmitglied beim Studierendenverband fzs, sagt: »Ist doch klar: Wenn ich mich für den Stoff interessiere und mir ein Seminar gefällt, gehe ich hin. Und dann engagiere ich mich und bekomme gute Noten.« Stimmt - sagt auch Hochschulforscher Schulmeister. Aber eben nicht nur: Natürlich sei die persönliche Motivation entscheidend sowohl für die Anwesenheit im Seminar als auch für das Selbststudium zu Hause und damit für den persönlichen Studienerfolg. Das zeigten fast alle einschlägigen Studien. Doch sei es ein gravierender Irrtum, daraus zu folgern, eine Erhöhung der Anwesenheitsquote mithilfe besonderer Belohnungen oder Strafen bringe nichts, weil die Motivation zur Anwesenheit ja dann nicht aus den Studenten selbst heraus komme. Schulmeister spricht von einer »besseren Lehrorganisation«. Man könnte es auch Kontrolle und Zwang nennen. »Das trägt zusätzlich zum Lernerfolg bei.« Schulmeister hat noch mehr herausgefunden: Ältere Studenten kommen regelmäßiger in die Veranstaltungen, und je schwieriger die Kurse werden, desto wichtiger ist die Anwesenheit für die Note. Desto häufiger fehlen allerdings auch die leistungsschwächeren Studenten.

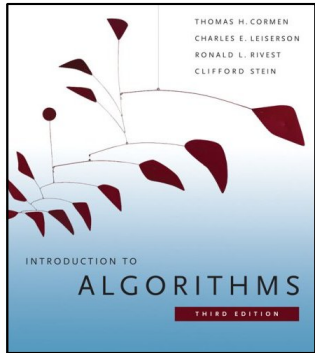
Svenja Schulze schwärmte von einem »Meilenstein«. Das neue Hochschulgesetz der rot-grünen Landesregierung bringe an den **Hochschulen** endlich wieder Freiheit und Verantwortung ins Gleichgewicht, sagte die nordrhein-westfälische Wissenschaftsministerin im Herbst 2014. Den Rektoren war weniger zum Feiern zumute: Sie fühlten sich in ihrer Freiheit beschränkt. Ganz im Gegensatz zu den Studenten: Ihnen brachte das Gesetz eine Unabhängigkeit, die sich Kommilitonen anderswo nur wünschen können. Bis auf wenige Ausnahmen keine Anwesenheitskontrolle mehr in den Vorlesungen und Seminaren, jeder kann so oft fehlen, wie er will. Die Begründung der Ministerin: »Die Studierenden sind Erwachsene. Die können selbst entscheiden, was gut für sie ist.«

Tatsächlich? Rolf Schulmeister, Hochschulforscher an der **Universität** Hamburg, war von Anfang an skeptisch, als er vom Ende der Anwesenheitspflicht hörte. NRW ist nicht das einzige Bundesland, das die Studenten in die Freiheit entlassen hat. Wohl aber dasjenige, so Schulmeister, das eine »denkwürdig merkwürdige« Kombination zweier Vorschriften im neuen Hochschulgesetz verankert habe. Vorschrift eins: »Die Hochschulen sind dem Studienerfolg verpflichtet.« Vorschrift zwei: besagtes Verbot von

DIE ZEIT Nr. 48/2015, 26.11.2015

<https://www.zeit.de/2015/48/anwesenheitspflicht-universitaet-schlechtere-leistung>

Literatur zu Algorithmen & Datenstrukturen



Cormen, Leiserson, Rivest, Stein:

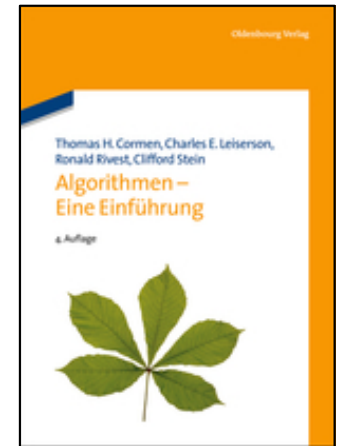
Introduction to Algorithms

MIT Press, 3. Aufl., 2009. Ca. 100 \$.

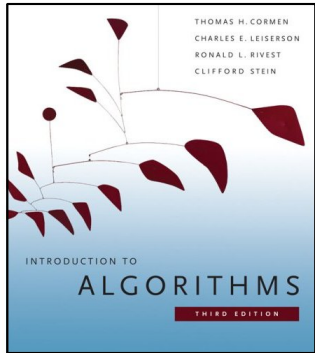
oder

Algorithmen – eine Einführung

De Gruyter Oldenbourg, 4. Aufl., 2017. Ca. 90 €.



Literatur zu Algorithmen & Datenstrukturen



Cormen, Leiserson, Rivest, Stein:

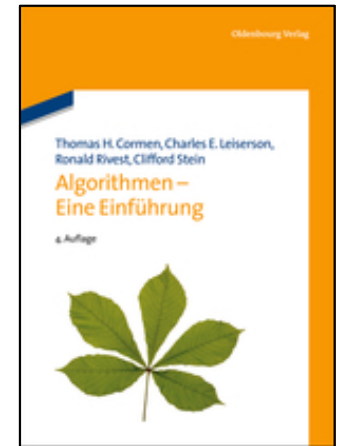
Introduction to Algorithms

MIT Press, 3. Aufl., 2009. Ca. 100 \$.

oder

Algorithmen – eine Einführung

De Gruyter Oldenbourg, 4. Aufl., 2017. Ca. 90 €.

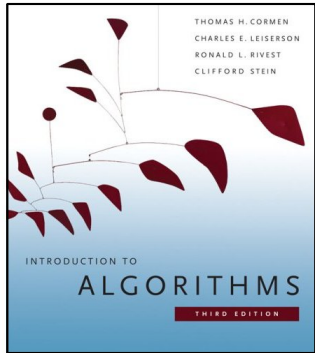


Ottmann & Widmayer:

Algorithmen und Datenstrukturen

Spektrum-Verlag, 5. Aufl., 2012. Ca. 50 €.

Literatur zu Algorithmen & Datenstrukturen



Cormen, Leiserson, Rivest, Stein:

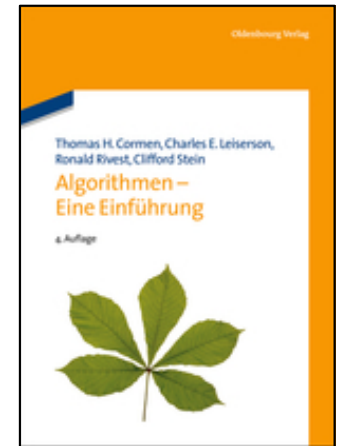
Introduction to Algorithms

MIT Press, 3. Aufl., 2009. Ca. 100 \$.

oder

Algorithmen – eine Einführung

De Gruyter Oldenbourg, 4. Aufl., 2017. Ca. 90 €.



Ottmann & Widmayer:

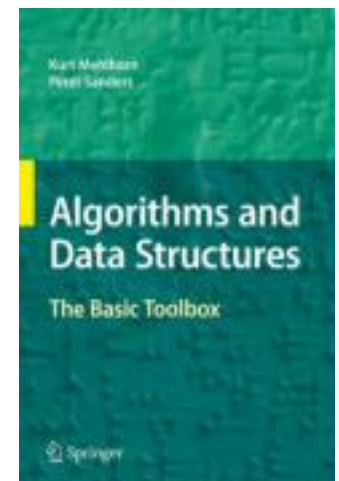
Algorithmen und Datenstrukturen

Spektrum-Verlag, 5. Aufl., 2012. Ca. 50 €.

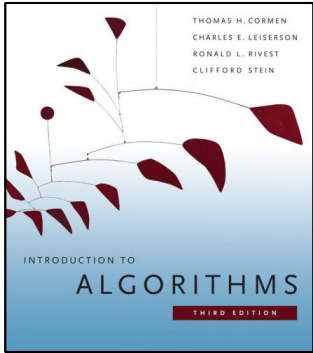
Mehlhorn & Sanders:

Algorithms and Data Structures: The Basic Toolbox

Springer, 2008. Ca. 38 €.



Literatur zu Algorithmen & Datenstrukturen



Cormen, Leiserson, Rivest, Stein:

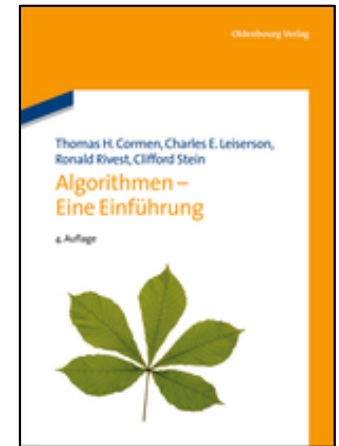
Introduction to Algorithms

MIT Press, 3. Aufl., 2009. Ca. 100 \$.

oder

Algorithmen – eine Einführung

De Gruyter Oldenbourg, 4. Aufl., 2017. Ca. 90 €.



Kleinberg & Tardos:

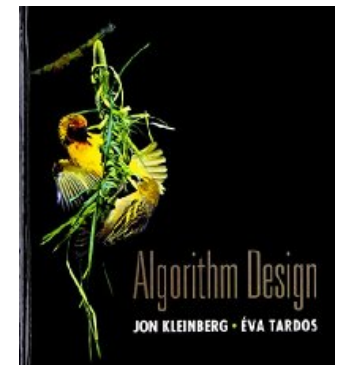
Algorithm Design

Pearson, 2006. Ca. 90 €.

Ottmann & Widmayer:

Algorithmen und Datenstrukturen

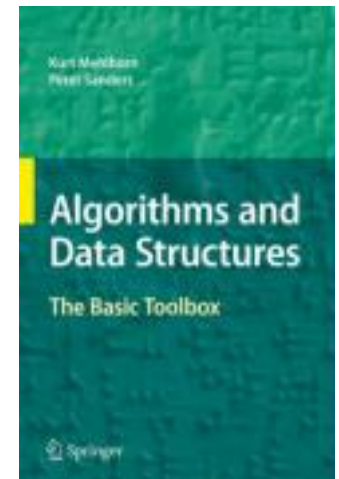
Spektrum-Verlag, 5. Aufl., 2012. Ca. 50 €.



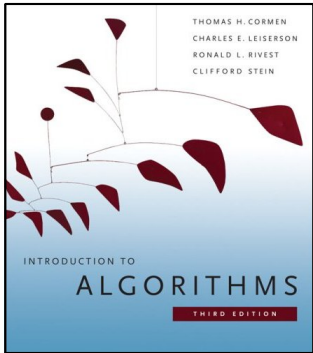
Mehlhorn & Sanders:

Algorithms and Data Structures: The Basic Toolbox

Springer, 2008. Ca. 38 €.



Literatur zu Algorithmen & Datenstrukturen



Cormen, Leiserson, Rivest, Stein:

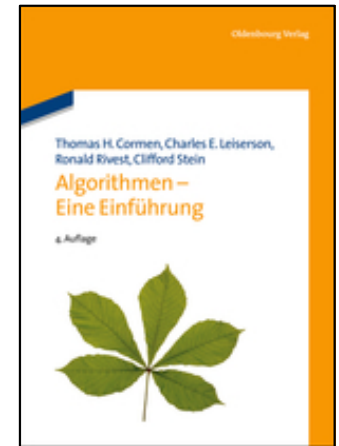
Introduction to Algorithms

MIT Press, 3. Aufl., 2009. Ca. 100 \$.

oder

Algorithmen – eine Einführung

De Gruyter Oldenbourg, 4. Aufl., 2017. Ca. 90 €.



Kleinberg & Tardos:

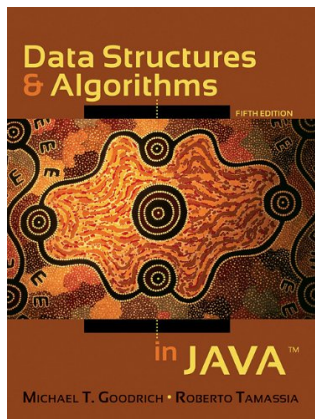
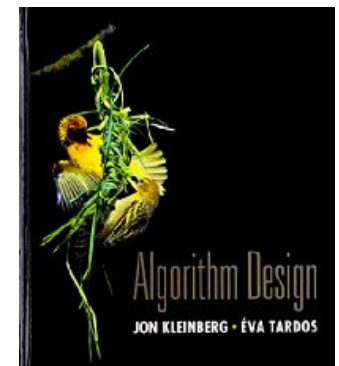
Algorithm Design

Pearson, 2006. Ca. 90 €.

Ottmann & Widmayer:

Algorithmen und Datenstrukturen

Spektrum-Verlag, 5. Aufl., 2012. Ca. 50 €.



Mehlhorn & Sanders:

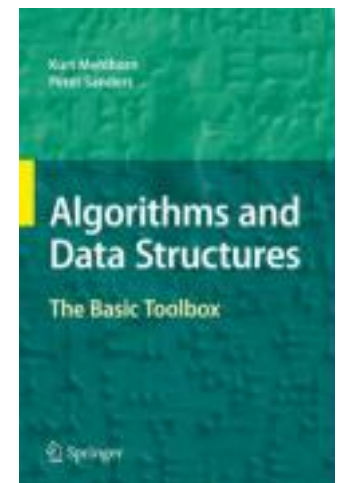
Algorithms and Data Structures: The Basic Toolbox

Springer, 2008. Ca. 38 €.

Goodrich & Tamassia:

Data Structures & Algorithms in Java.

Wiley, 5. Aufl., 2010. Ca. 115 €.



Literatur über Java

- D. Ratz, J. Scheffler, D. Seese, J. Wiesenberger:
Grundkurs Programmieren in Java (8. Aufl.)

Hanser Verlag

<http://www.grundkurs-java.de/>

- C. Ullenboom:
Java ist auch eine Insel

Galileo Computing

openbook.galileocomputing.de/javainsel/



Literatur über Java

- D. Ratz, J. Scheffler, D. Seese, J. Wiesenberger:
Grundkurs Programmieren in Java (8. Aufl.)

Hanser Verlag

<http://www.grundkurs-java.de/>



- C. Ullenboom:
Java ist auch eine Insel

Galileo Computing

openbook.galileocomputing.de/javainsel/



- Für alle, die Java noch nicht kennen *und* nicht beim Vorkurs waren:

Literatur über Java

- D. Ratz, J. Scheffler, D. Seese, J. Wiesenberger:
Grundkurs Programmieren in Java (8. Aufl.)

Hanser Verlag

<http://www.grundkurs-java.de/>



- C. Ullenboom:
Java ist auch eine Insel

Galileo Computing

openbook.galileocomputing.de/javainsel/



- Für alle, die Java noch nicht kennen *und* nicht beim Vorkurs waren:

→ WueCampus-Kurs „Programmiervorkurs“ (WS 2022/23):

<https://wuecampus2.uni-wuerzburg.de/moodle/enrol/index.php?id=54425>

Arbeiten Sie insbesondere alle Übungsaufgaben durch!

TO DO

16
FIRST
THINGS
FIRST

TO DO

- Erfüllen Sie die Voraussetzungen?

$$\frac{\log_b x}{\log_b y} = ?$$

*Lesen Sie **Anhang A** im Buch von Cormen et al.!*
Lösen Sie die Übungsaufgaben dazu!

FIRST THINGS FIRST

TO DO

FIRST THINGS FIRST

- Erfüllen Sie die Voraussetzungen? $\frac{\log_b x}{\log_b y} = ?$

*Lesen Sie **Anhang A** im Buch von Cormen et al.!*
Lösen Sie die Übungsaufgaben dazu!

- *Schreiben Sie sich ein!*
 - **Vorlesung**sfolien, Übungsblätter, allgemeine Informationen:
<https://wuecampus2.uni-wuerzburg.de> „WueCampus“
 - Übungseinteilung **bis heute, 23:59 Uhr – bitte mit 3 Prioritäten**:
<https://wuestudy.zv.uni-wuerzburg.de/> „WueStudy“
 - **Chat**: Inhaltliche Fragen zur Vorlesung
<https://chat.uni-wuerzburg.de/invite/TZFubc> →
<https://chat.uni-wuerzburg.de/group/ads22>

TO DO

FIRST THINGS FIRST

- Erfüllen Sie die Voraussetzungen? $\frac{\log_b x}{\log_b y} = ?$

*Lesen Sie **Anhang A** im Buch von Cormen et al.!*
Lösen Sie die Übungsaufgaben dazu!

- *Schreiben Sie sich ein!*
 - **Vorlesung**sfolien, Übungsblätter, allgemeine Informationen:
<https://wuecampus2.uni-wuerzburg.de> „WueCampus“
 - Übungseinteilung **bis heute, 23:59 Uhr – bitte mit 3 Prioritäten**:
<https://wuestudy.zv.uni-wuerzburg.de/> „WueStudy“
 - **Chat**: Inhaltliche Fragen zur Vorlesung
<https://chat.uni-wuerzburg.de/invite/TZFubc> →
<https://chat.uni-wuerzburg.de/group/ads22>

TO DO

FIRST THINGS FIRST

- Erfüllen Sie die Voraussetzungen? $\frac{\log_b x}{\log_b y} = ?$

*Lesen Sie **Anhang A** im Buch von Cormen et al.!*
Lösen Sie die Übungsaufgaben dazu!

- *Schreiben Sie sich ein!*
 - **Vorlesung**sfolien, Übungsblätter, allgemeine Informationen:
<https://wuecampus2.uni-wuerzburg.de> „WueCampus“
 - Übungseinteilung **bis heute, 23:59 Uhr – bitte mit 3 Prioritäten**:
<https://wuestudy.zv.uni-wuerzburg.de/> „WueStudy“
 - **Chat**: Inhaltliche Fragen zur Vorlesung
<https://chat.uni-wuerzburg.de/invite/TZFubc> →
<https://chat.uni-wuerzburg.de/group/ads22>
- Installieren Sie vor Ihrer Übung
 - das Java Development Kit (neuste Version):
<https://www.oracle.com/java/technologies/downloads/>
 - die Java-Entwicklungsumgebung IntelliJ IDEA (neuste Version):
<https://www.jetbrains.com/idea/download> (→ community)