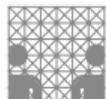




Universität Hamburg

DER FORSCHUNG | DER LEHRE | DER BILDUNG

MIN-Fakultät  
Fachbereich Informatik



# 64-040 Modul InfB-RSB

## Rechnerstrukturen und Betriebssysteme

[https://tams.informatik.uni-hamburg.de/  
lectures/2024ws/vorlesung/rsb](https://tams.informatik.uni-hamburg.de/lectures/2024ws/vorlesung/rsb)

– Info zur Organisation –

Andreas Mäder



Universität Hamburg  
Fakultät für Mathematik, Informatik und Naturwissenschaften  
Fachbereich Informatik

**Technische Aspekte Multimodaler Systeme**

Wintersemester 2024/2025



## 1. Rechnerstrukturen und Betriebssysteme

## 2. Organisatorisches

Vorlesung

Übungen

Tutorien

Praktika

Klausur

## 3. Literaturempfehlungen

## 4. Software





## Modulhandbuch

*Dieses Modul behandelt im Rahmen der Vorlesung ein begrenztes und wohl ausgewähltes Theorie- und Methodenrepertoire für die Konfigurierung, den Entwurf, die Realisierung und die angemessene Nutzung von Rechnern unter Berücksichtigung ihrer Basiskomponenten und der eingesetzten Betriebssysteme. Hierbei finden technologische, ökonomische und anwendungsspezifische Randbedingungen Berücksichtigung.*

*... das Verstehen der grundlegenden Konzepte, Organisationsformen und Entwurfsmethoden von Rechnersystemen und deren Vernetzung, einschließlich der Betriebs(system)software ...*

[www.inf.uni-hamburg.de/de/studies/bachelor](http://www.inf.uni-hamburg.de/de/studies/bachelor)

1. FS	Software-entwicklung I (6 LP)	Informatik im Kontext (6 LP)	Rechnerstrukturen und Betriebssysteme (9 LP)	Diskrete Mathematik für Studierende der Informatik (9 LP)		
2. FS	Software-entwicklung II (6 LP)	Verteilte Systeme u. Systemsicherheit (6 LP)	Pro-seminar (3 LP)	Einführung in die theoretische Informatik (6 LP)	Analysis/Lineare Algebra für Studierende der Informatik (9 LP)	
3. FS	Software Engineer. Einf. (3 LP)	Wahlpflicht 18 LP			Meth. komp. (3 LP)	Algorithmen und Datenstrukturen (6 LP)
4. FS	Wahlpflicht 12 LP		Praktikum (6 LP)	Berechenbarkeit, Komplexität und Approximation (6 LP)	Stochastik I (6 LP)	
5. FS	Wahlpflicht 9 LP	Wahlbereich 9 LP	Projekt (9 LP)	Seminar (3 LP)		
6. FS	Wahlpflicht 9 LP	Wahlbereich 9 LP	Abschlussmodul (Bachelorarbeit, 12 LP)			

# Software-System-Entwicklung BSc

Rechnerstrukturen und Betriebssysteme

Organisatorisches

Literaturempfehlungen

Software

1. FS	Software-entwicklung I (6 LP)	Informatik im Kontext (6 LP)	Rechnerstrukturen und Betriebssysteme (9 LP)	Diskrete Mathematik für Studierende der Informatik (9 LP)		
2. FS	Software-entwicklung II (6 LP)	Verteilte Systeme u. Systemsicherheit (6 LP)	Proseminar (3 LP)	Einführung in die theoretische Informatik (6 LP)	Analysis/Lineare Algebra für Studierende der Informatik (9 LP)	
3. FS	Software Engineer. Einf. (3 LP)	Grundlagen von Datenbanken (6 LP)	Projektmanag. (3 LP)	Meth. komp. (3 LP)	Wahlpflicht 9 LP	Algorithmen und Datenstrukturen (6 LP)
4. FS	Softwareentwurf (6 LP)	Wahlpflicht 6 LP	Wahlbereich 3 LP	Software-entwicklungspraktikum (6 LP)	Projekt (9 LP)	
5. FS	Wahlbereich 9 LP	Seminar (3 LP)	Industriepraktikum (18 LP)			
6. FS	Wahlpflicht 9 LP	Wahlbereich 9 LP	Abschlussmodul (Bachelorarbeit, 12 LP)			



## Einführende Grundlagen

- ▶ Grundbegriffe der Informationsverarbeitung:  
Information, Repräsentation von Information, Zahlensysteme, Codierung
- ▶ Rechnerarithmetik:  
Operationen mit Zahlen, Ganzzahl- und Gleitkomma-Arithmetik
- ▶ Konzepte der Digitaltechnik:  
Boole'sche Algebra, Schaltnetze, Automatentheorie, synchrone und asynchrone Schaltwerke, gekoppelte Schaltwerke (Operationswerk, Steuerwerk)
- ▶ Hardwarestrukturen:  
Register, Registerbank, Zähler, Speicher, Datenpfade (ALUs), Bus-Systeme



## Rechnerarchitektur

- ▶ von-Neumann-Rechner: Prinzipien, Architektur, Befehlssatz
- ▶ ISA-Konzepte: CISC, RISC
- ▶ Maschinen- und Assembler-Code und -programmierung
- ▶ Pipelining, superskalare Architekturen, Parallelrechner
- ▶ Speicherhierarchie: Speichertypen, Cache
- ▷ IC-Schaltungstechnik, VLSI-Entwurf und -Fertigung

## Betriebssysteme

- ▶ Konzepte und Aufgaben
- ▶ Ressourcenverwaltung, Scheduling
- ▶ Speicherhierarchie: virtueller Speicher
- ▷ Prozesskommunikation, Locking
- ▷ Dateisysteme, Schnittstellen und Peripheriegeräte
- ▷ Sicherheitsaspekte



- ▶ Zwischenfragen bei Unklarheiten etc.!
- ▶ Fehler und Ungenauigkeiten in den Folien und Materialien bitte melden
- ▶ Vorschläge und Hinweise auf Tools, Lehrmaterialien etc. sind immer willkommen!
- ▶ Lehrevaluation: Umfrage am Ende des Semesters Kritikpunkte, aber auch gerne Positives

Problem: stark unterschiedliches Vorwissen!

- ▶ generell: keine speziellen Voraussetzungen
  - ▶ betrifft insbesondere **Rechnerstrukturen und Betriebssysteme**
- ⇒ Geduld, wenn (am Anfang) Stoff schon bekannt
- ⇒ kein Frust, wenn sehr speziell, sondern: Eigeninitiative (Zwischenfragen, Tutorium, Übungsgruppen ...)



Dr. Andreas Mäder

`andreas.maeder@uni-hamburg.de`

+49 40 42883 2502

Informatikum, Haus F 317





- ▶ Do. 08:30-10:00 Audimax 2
- ▶ Fr. 14:15-15:45 Erzwiss. H (4 SWS)

## Materialien

- ▶ Moodle [lernen.min.uni-hamburg.de](https://lernen.min.uni-hamburg.de)
  - ▶ lokale Webseiten [tams.informatik.uni-hamburg.de/lectures/2024ws/vorlesung/rsb](https://tams.informatik.uni-hamburg.de/lectures/2024ws/vorlesung/rsb)
  - ▶ STiNE [www.stine.uni-hamburg.de](https://www.stine.uni-hamburg.de)
- ⇒ werden alle aktualisiert

- ▶ Diskussionsforum Q & A (Questions and Answers) im Moodle als Dokumentation zum „Nachlesen“
- ▶ Folien (pdf) kapitelweise zum Download, Überarbeitung während des Semesters
- ▶ diverse gute Lehrbücher — Empfehlungen s.u.
- ▶ Software: JAVA VM, C-Compiler (GNU-Toolchain)



- ▶ 17 Gruppen à 18 Studierende
- ▶ 1 SWS, wöchentlich
- ▶ Beginn: nächste Woche! (ab 21.10. KW43)
  
- ▶ Diskussion und Nachbereitung der Vorlesung
- ▶ Gruppenarbeit: 2-3 Studies pro Gruppe
  
- ▶ Übungsaufgaben zum Vertiefen und Erarbeiten des Stoffes
- ▶ Aufgabenblätter jeweils Mittwochs zum Download verfügbar
- ▶ Abgabe der Lösungen: E-Mail bis nächsten Mittwoch 24:00 direkt an Tutorin/Tutor
  
- ▶ Vorlesung: 284    Übungen: 255

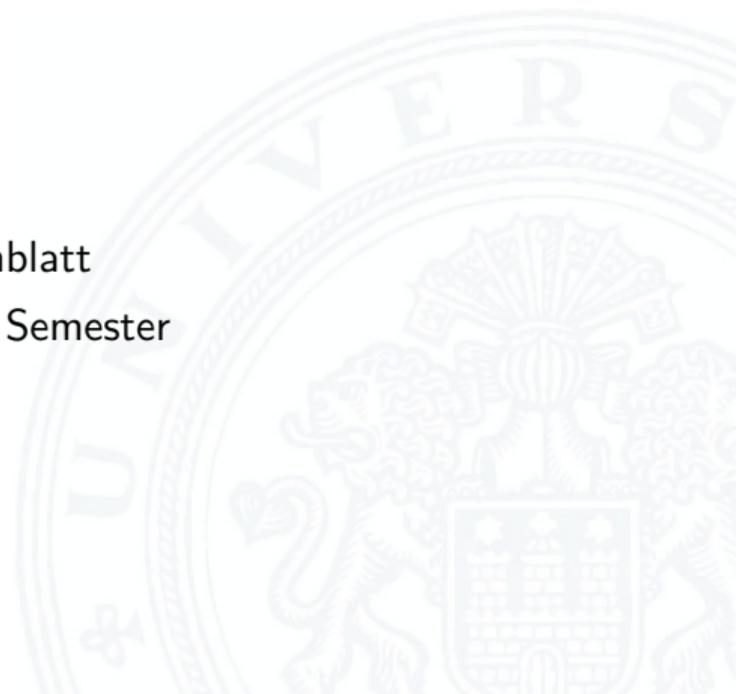
[STINE: 14.10.]



# Übungen: Scheinkriterien

[tams.informatik.uni-hamburg.de/lectures/2024ws/vorlesung/rsb/doc/scheinkriterien.pdf](https://tams.informatik.uni-hamburg.de/lectures/2024ws/vorlesung/rsb/doc/scheinkriterien.pdf)

- ▶ regelmäßige aktive Teilnahme
- ▶ mindestens zweimal in der Gruppe vorrechnen
- ▶ höchstens zweimal (unentschuldigt) gefehlt
  
- ▶ alle Aufgabenblätter bearbeitet
- ▶ jeweils mindestens 30 % der Punkte pro Aufgabenblatt
- ▶ und mindestens 50 % der Gesamtpunktzahl im Semester





- ▶ freiwillig, Nacharbeiten von Stoff und Übungen
- ▶ Diskussion, Fragen: mehr Zeit als in der Übung

1. ab nächster Woche (24. Okt.): **Do., ab 18:30, Ikum G 021/022**

+ BBB Live-Streaming und -Chat: [bbb1.physnet.uni-hamburg.de/b/jes-7pm-ubg-hqm](https://bbb1.physnet.uni-hamburg.de/b/jes-7pm-ubg-hqm)

+ YouTube Kanal: [www.youtube.com/channel/UC10j3PauhVSeAUgC39VYMsA](https://www.youtube.com/channel/UC10j3PauhVSeAUgC39VYMsA)  
lokale Aufzeichnung des Tutoriums

2. Vorbereitung auf die Klausur: „*Paniktutorium*“

Termin: kurz vor der Klausur, Modus wird noch bekanntgegeben



- ▶ bei Informatik Nebenfach z.T. freiwillig: Studienplan beachten!
- ▶ 16 Kurse à 15-18 Studierende, jeweils 2-3 Tutoren (Betreuungsaufwand)
  - 2 Online-Kurse      Blockkurse: 4 Termine jeweils 3 h (1 SWS)
  - 12 × 4-wöchig: 1 Tag, 3 h – im Semester    Nov.+Jan.
  - 3 × 1-wöchig: 4 Tage, 3 h – VL-freie Zeit    Feb.+März
- ▶ Vorbereitung **unbedingt** erforderlich
- ▶ **individuelle Abnahme** der Ergebnisse

Thema: Erarbeiten und Programmieren einer einfachen Prozessorarchitektur

- ▶ Komponenten auf der Register-Transfer-Ebene
- ▶ Zeitverhalten, Speicheransteuerung
- ▶ Mikroprogrammierung
- ▶ Assemblerprogrammierung
- ▶ I/O-Operationen und Interrupts



- ▶ Abschlussprüfung des Moduls Rechnerstrukturen und Betriebssysteme
- ▶ Note geht in BSc-Zeugnis ein

- ▶ Nachteilsausgleich

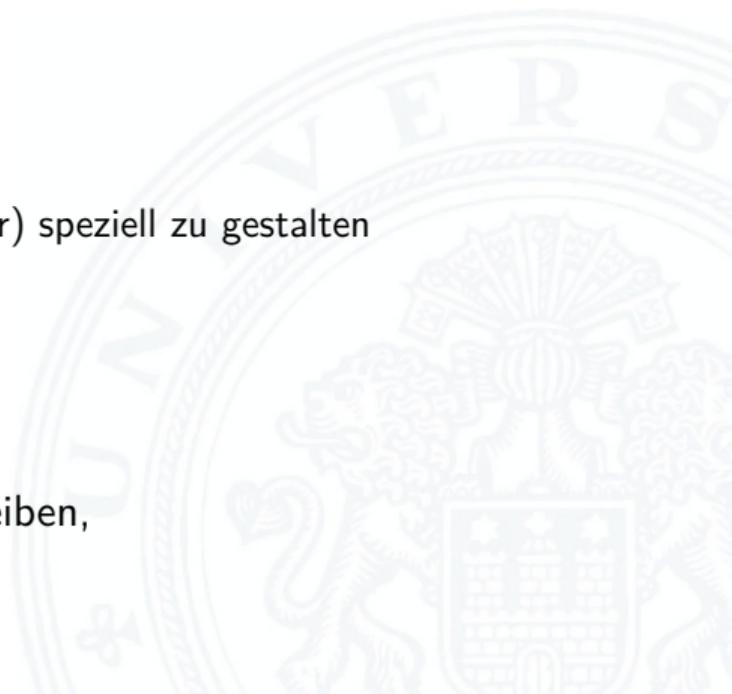
- ▶ bei mir melden!
- ▶ zahlreiche Möglichkeiten Prüfungsleistung (Klausur) speziell zu gestalten

- ▶ zwei Klausurtermine in den Semesterferien

Di. 04. Feb. 2025 16:30-18:30 Audimax 1 & 2

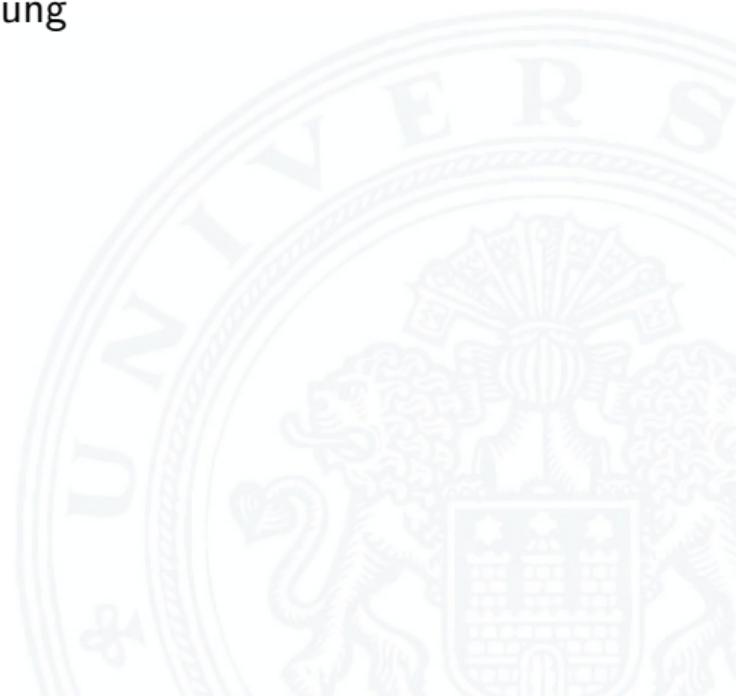
Di. 11. März 2025 09:30-11:30 ESA A & B

- ▶ Tipp: möglichst bereits die erste Klausur mitschreiben, ggf. Wiederholung direkt zum zweiten Termin





- ▶ Umfang und Ablauf wie „echte“ Klausur
- ▶ 2-stündige Klausur
  - + anschließende gemeinsame Korrektur/Besprechung
- ▶ Termin: letzte Semesterwoche?





[BO15] R.E. Bryant, D.R. O'Hallaron:

*Computer systems – A programmers perspective.*

3rd global ed., Pearson Education Ltd., 2015.

ISBN 978-1-292-10176-7

[csapp.cs.cmu.edu](http://csapp.cs.cmu.edu)

Rechnerarchitektur mit Schwerpunkt Software und Systeme, leider nicht ganz billig.  
Viele C-Programme und Systemprogrammierung. Beispiele anhand Intel x86 Architektur.

[TA14] A.S. Tanenbaum, T. Austin:

*Rechnerarchitektur – Von der digitalen Logik zum Parallelrechner.*

6. Auflage, Pearson Deutschland GmbH, 2014.

ISBN 978-3-86894-238-5

Guter Überblick, klares didaktisches Konzept. Java VM, Intel x86, SPARC.  
Mit jeder Auflage komplett überarbeitet und aktualisiert.



[Sta17] W. Stallings:

*Operating Systems – Internals and Design Principles.*

9th global ed., Pearson Education, 2017.

ISBN 978-1-292-21429-0

Guter Überblick; auch aktuellere Betriebssysteme werden angesprochen.

[Bau22] C. Baun: *Betriebssysteme kompakt – Grundlagen, Hardware, Speicher, Daten und Dateien, Prozesse und Kommunikation, Virtualisierung.*

3. Auflage, Springer-Verlag GmbH, 2022.

ISBN 978-3-662-64718-9

Kompakt, deshalb fehlen aber vertiefende Inhalte und Beispiele von Betriebssystemen.



- [PH22] D.A. Patterson, J.L. Hennessy: *Rechnerorganisation und Rechnerentwurf – Die Hardware/Software-Schnittstelle – MIPS Edition*.  
6. Auflage, De Gruyter Oldenbourg, 2022. ISBN 978-3-11-075598-5  
Schönes Lehrbuch von den Entwicklern der RISC/MIPS Prozessoren.
- [PH20] D.A. Patterson, J.L. Hennessy: *Computer Organization and Design – The Hardware Software Interface – RISC-V Edition*.  
2nd edition, Morgan Kaufmann Publishers Inc., 2020. ISBN 978-0-12-820331-6  
... auch mit RISC-V Befehlssatz (oder als „ARM Edition“).
- [Sta21] W. Stallings:  
*Computer Organization and Architecture – Designing for Performance*.  
11th global ed., Pearson Education Ltd., 2021. ISBN 978-1-292-42010-3

- [SS04] W. Schiffmann, R. Schmitz:  
*Technische Informatik 1 – Grundlagen der digitalen Elektronik.*  
5. Auflage, Springer-Verlag, 2004. ISBN 978-3-540-40418-7
- [Sch05] W. Schiffmann:  
*Technische Informatik 2 – Grundlagen der Computertechnik.*  
5. Auflage, Springer-Verlag, 2005. ISBN 978-3-540-22271-2
- [SBH11] W. Schiffmann, H. Bähring, U. Hönig:  
*Technische Informatik 3 – Grundlagen der PC-Technologie.*  
Springer-Verlag, 2011. ISBN 978-3-642-16811-6
- [SSW04] W. Schiffmann, R. Schmitz, J. Weiland:  
*Übungsbuch zur Technische Informatik 1 und 2.*  
3. Auflage, Springer-Verlag, 2004. ISBN 978-3-540-20793-1  
Dutzende von Übungsaufgaben mit detailliert entwickelten Lösungen.



- [Mu<sup>+</sup>09] T. Müller [u. a.]: *Technische Informatik I – Grundlagen der Informatik und Assemblerprogrammierung*.  
3. Auflage, vdf, 2009. ISBN 978-3-7281-3255-0
- [Gu<sup>+</sup>10] R. Gübeli [u. a.]: *Technische Informatik II – Mikroprozessor-Hardware und Programmier Techniken*.  
2. Auflage, vdf, 2010. ISBN 978-3-7281-3256-7
- [MH07] M.J. Murdocca, V.P. Heuring:  
*Computer architecture and organization – An integrated approach*.  
John Wiley & Sons, 2007. ISBN 978-0-471-73388-1



[SGG18] A. Silberschatz, P.B. Galvin, G. Gagne:

*Operating System Concepts.*

10th edition, John Wiley & Sons, 2018.

ISBN 978-1-119-32091-3

Ein Klassiker. . . Beispiele von Betriebssystemen leider nicht mehr aktuell.

[TB23] A.S. Tanenbaum, H. Bos:

*Modern Operating Systems.*

5th Global ed., Pearson, 2023.

ISBN 978-1-292-72789-9

Sehr gutes Buch, in der 5. Auflage auch mit aktuellen Beispielen von Betriebssystemen.

[Bra17] R. Brause:

*Betriebssysteme – Grundlagen und Konzepte.*

4. Auflage, Springer-Verlag GmbH, 2017.

ISBN 978-3-662-54099-2



[HP17] J.L. Hennessy, D.A. Patterson:

*Computer architecture – A quantitative approach.*

6th edition, Morgan Kaufmann Publishers Inc., 2017. ISBN 978-0-12-811905-1

Die Bibel zum Thema Rechnerarchitektur!

[Knu05] D.E. Knuth: *The Art of Computer Programming, Volume 1, Fascicle 1, MMIX – A RISC Computer for the New Millennium.*

Addison-Wesley Professional, 2005.

ISBN 978-0-201-85392-6

[www-cs-faculty.stanford.edu/~knuth/taocp.html](http://www-cs-faculty.stanford.edu/~knuth/taocp.html)

[Knu08] D.E. Knuth: *The Art of Computer Programming, Volume 4, Fascicle 0, Introduction to Combinatorial Algorithms and Boolean Functions.*

Addison-Wesley Professional, 2008.

ISBN 978-0-321-53496-5



[Knu09] D.E. Knuth: *The Art of Computer Programming, Volume 4, Fascicle 1, Bitwise Tricks & Techniques; Binary Decision Diagrams.*

Addison-Wesley Professional, 2009.

ISBN 978-0-321-58050-4

[DM94] G. De Micheli:

*Synthesis and Optimization of Digital Circuits.*

Mc Graw-Hill, 1994.

ISBN 978-0-07-016333-1

[si2.epfl.ch/demichel/publications/mcgraw](http://si2.epfl.ch/demichel/publications/mcgraw)





[Lag87] K. Lagemann: *Rechnerstrukturen*.  
Springer-Verlag, 1987.

ISBN 3-540-17618-7

[Möl03] D.P. Möller: *Rechnerstrukturen: Grundlagen der Technischen Informatik*.  
Springer-Verlag, 2003.

ISBN 3-540-67638-4

[Mäd11] A. Mäder: *Vorlesung: Rechnerarchitektur und Mikrosystemtechnik*.  
Universität Hamburg, FB Informatik, 2011, Vorlesungsfolien.  
[tams.informatik.uni-hamburg.de/lectures/2011ws/vorlesung/ram](http://tams.informatik.uni-hamburg.de/lectures/2011ws/vorlesung/ram)



[Hyd10] R. Hyde: *The Art of Assembly Language*.

2nd edition, No Starch Press, 2010.

ISBN 978-1-5932-7207-4

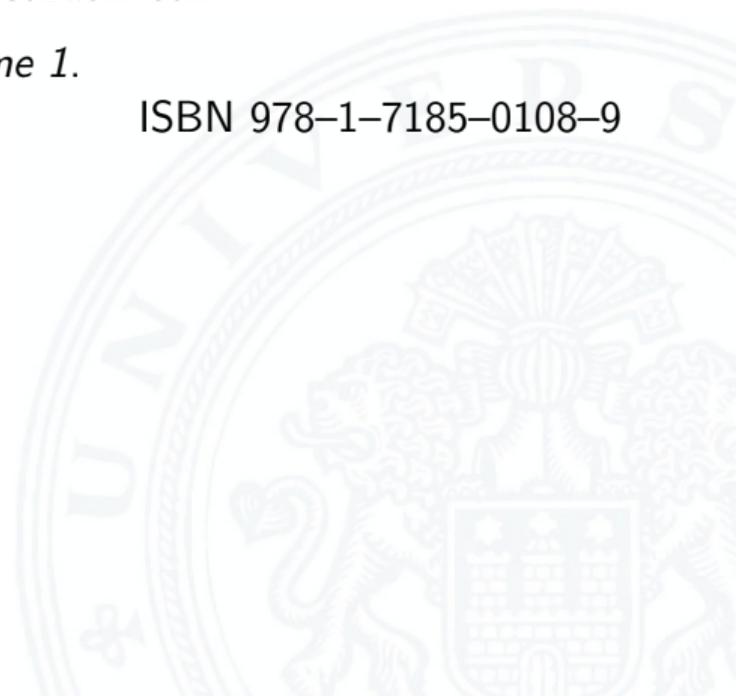
[www.randallhyde.com/AssemblyLanguage/](http://www.randallhyde.com/AssemblyLanguage/)[www.artofasm.com](http://www.artofasm.com)

[Hyd21] R. Hyde: *The Art of 64-bit Assembly, Volume 1*.

No Starch Press, 2021.

ISBN 978-1-7185-0108-9

[artofasm.randallhyde.com](http://artofasm.randallhyde.com)





- ▶ Java VM für diverse Programmierbeispiele
- ▶ HADES Schaltungssimulator — HAMBURG DEsign System  
`tams.informatik.uni-hamburg.de/applets/hades/webdemos`
- ▶ C-Compiler, Assembler und Tools, Debugger
- ▶ GNU-Toolchain empfehlenswert (gcc, binutils, gdb...)
  - Linux**
    - ▶ auf Informatik Pool-PCs installiert
    - ▶ direkte Installation in Distribution
  - Windows**
    - ▶ Cygwin mit Devel-Tools installieren ([www.cygwin.com](http://www.cygwin.com))
    - ▶ Linux VM mit GNU-Toolchain (VirtualBox, VMware, etc.)
    - ▶ Windows-Subsystem für Linux (WSL)
- ▶ Links und weitere Infos im Moodle oder auf der Webseite zur Vorlesung