



64-040 Modul InfB-RSB

Rechnerstrukturen und Betriebssysteme

[https://tams.informatik.uni-hamburg.de/
lectures/2020ws/vorlesung/rsb](https://tams.informatik.uni-hamburg.de/lectures/2020ws/vorlesung/rsb)

– Info zur Organisation –

Andreas Mäder



Universität Hamburg
Fakultät für Mathematik, Informatik und Naturwissenschaften
Fachbereich Informatik

Technische Aspekte Multimodaler Systeme

Wintersemester 2020/2021



1. Rechnerstrukturen und Betriebssysteme

2. Organisatorisches

Digitale Lehre

Vorlesung

Übungen

Tutorien

Praktikum

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

1. FS	Software-entwicklung I	Diskrete Mathematik für Studierende der Informatik	Rechnerstrukturen und Betriebssysteme	Informatik im Kontext	
2. FS	Software-entwicklung II	Analysis/Lineare Algebra für Studierende der Informatik	Proseminar	Verteilte Systeme und Systemsicherheit	Einführung in die theoretische Informatik
3. FS	Algorithmen und Datenstrukturen	Wahlpflicht 21 LP			Meth. komp.
4. FS	Berechenbarkeit, Komplexität und Approximation	Stochastik I	Wahlpflicht 12 LP	Praktikum	
5. FS	Wahlpflicht 9 LP	Wahl 9 LP	Projekt	Seminar	
6. FS	Wahlpflicht 9 LP	Wahl 9 LP	Abschlussmodul (Bachelorarbeit)		

Software-System-Entwicklung BSc

Rechnerstrukturen und Betriebssysteme

Organisatorisches

Literaturempfehlungen

Software

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

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,
kooperierende Schaltwerke (Operationswerk, Steuerwerk)
- ▶ Hardwarestrukturen:
Register, Registerbank, Zähler, Speicher,
Datenpfade, 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



Feedback erwünscht

maeder@informatik.uni-hamburg.de

- ▶ Zwischenfragen: bitte Feedback bei Unklarheiten etc.!
- ▶ Fehler und Ungenauigkeiten in den Folien und Materialien bitte melden
- ▶ Vorschläge und Hinweise auf Tools, Lehrmaterialien etc. sind immer willkommen!

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 (Fragen, Tutorium, Übungsgruppen ...)



Dr. Andreas Mäder

`maeder@informatik.uni-hamburg.de`

+49 40 42883 2502

Informatikum, Haus F-317



wegen Corona-Virus

- ▶ aufwändige Planung: Vorlesung in Teilpräsenz, wechselnde Präsenz-Übungsgruppen etc.
- ▶ wichtig für Erstsemester:
universitäres Arbeiten in Übungs- und Lerngruppen...
- ▶ jetzt: **nur digitale Lehre**

digitale Lehre

- ▶ Moodle lernen.min.uni-hamburg.de
 - ▶ lokale Webseiten
tams.informatik.uni-hamburg.de/lectures/2020ws/vorlesung/rsb
 - ▶ STiNE www.stine.uni-hamburg.de
 - ▶ Lecture2Go lecture2go.uni-hamburg.de/l2gos
- ⇒ werden alle aktualisiert

- ▶ Lecture2Go Videos im Offline-Streaming, ca. 3h pro Woche
 - ▶ Videokonferenz: **Freitag, ab 12:15**
Fragen zu aktuellen Vorlesungsinhalten
 - ▶ Diskussionsforum Q & A (Questions and Answers)
Dokumentation zum „Nachlesen“
 - ▶ Folien (pdf) kapitelweise zum Download
teilweise Überarbeitung während des Semesters
 - ▶ diverse gute Lehrbücher — Empfehlungen s.u.
- ⇒ Informationen und Downloads im Moodle — **aktuell!**
- ▶ Software: JAVA VM, C-Compiler, GNU-Toolchain

Übungen

tams.informatik.uni-hamburg.de/lectures/2020ws/vorlesung/rsb/uebung

- ▶ 20 Gruppen, kleine Größe wegen Online
- ▶ 1 SWS, wöchentlich
- ▶ Beginn: nächste Woche!

- ▶ Diskussion und Nachbereitung der Vorlesung
- ▶ Gruppenarbeit: 2-3 Teilnehmer 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 beim Gruppenleiter

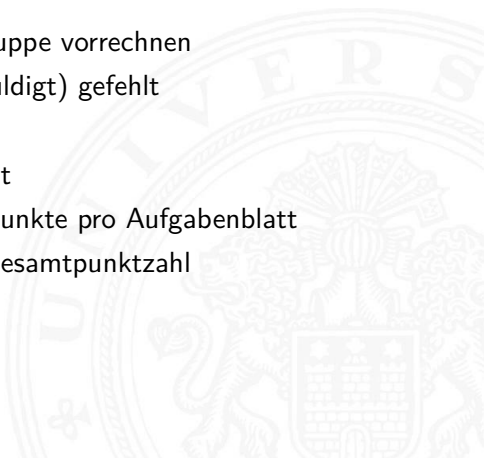
- ▶ Vorlesung: 307 Übungen: 280

[STiNE: 03.11.]



- ▶ tams.informatik.uni-hamburg.de/lectures/2020ws/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



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

1. ab nächster Woche: **Donnerstag, ab 18:30**

+ BBB Live-Streaming und -Chat:

bbb1.physnet.uni-hamburg.de/b/jes-7pm-ubg-hqm

+ YouTube Kanal:

www.youtube.com/channel/UC10j3PauhVSeAUgC39VYMsa

lokale Aufzeichnung des Tutoriums

2. Vorbereitung auf die Klausur: „*Paniktutorium*“
Termin: kurz vor der Klausur, Modus wird noch bekanntgegeben

- ▶ 24 Kurse á 16, bzw. 8 Teilnehmer (z.T. doppelt)
sehr Betreuungsaufwändig: viele Fragen, Diskussion!
- ▶ Vorbereitung **unbedingt** erforderlich

- ▶ Erarbeiten und Programmieren eines einfachen Prozessors
- ▶ 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
- ▶ Termine und Art der Durchführung noch unklar!
gilt auch für „Probeklausur“

[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

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 978–3–8689–4238–5

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.

[Bau20] C. Baun: *Betriebssysteme kompakt – Grundlagen, Daten, Speicher, Dateien, Prozesse und Kommunikation*.
2. Auflage, Springer-Verlag GmbH, 2020.
ISBN 978–3–662–61410–5

Sehr kompakt, deshalb fehlen aber vertiefende Inhalte und Beispiele.

[PH20] D.A. Patterson, J.L. Hennessy: *Computer Organization and Design – The Hardware Software Interface – MIPS Edition*.
6th edition, Morgan Kaufmann Publishers Inc., 2020.

ISBN 978–0–12–820109–1

Schönes Lehrbuch von den Entwicklern der RISC/MIPS Prozessoren.

[PH16] D.A. Patterson, J.L. Hennessy: *Rechnerorganisation und Rechnerentwurf – Die Hardware/Software-Schnittstelle*.
5. Auflage, De Gruyter Oldenbourg, 2016. ISBN

978–3–11–044605–0

Die deutsche Übersetzung; entspricht der 2016 „ARM Edition“.

[Sta19] W. Stallings: *Computer Organization and Architecture – Designing for Performance*.
11th edition, Pearson International, 2019.

ISBN 978–0–13–499719–3

- [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⁺09] T. Müller [u. a.]: *Technische Informatik I – Grundlagen der Informatik und Assemblerprogrammierung*. 3. Auflage, vdf, 2009. ISBN 978-3-7281-3255-0
- [Gu⁺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... in neuer Auflage auch mit aktuellen Beispielen.

[TB16] A.S. Tanenbaum, H. Bos: *Moderne Betriebssysteme.*

4. Auflage, Pearson Deutschland GmbH, 2016.

ISBN 978-3-8689-4270-5

Sehr gutes Buch, leider mit alten 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



- [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 0-07-016333-2.
si2.epfl.ch/~demichel/publications/mcgraw



[Lag87] K. Lagemann: *Rechnerstrukturen*.

Springer-Verlag, 1987. ISBN 3-540-17618-7

[Mö103] 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



[Hyd10] R. Hyde: *The Art of Assembly Language Programming*.
2nd edition, No Starch Press, 2010. ISBN 978-1-59327-207-4.
www.plantation-productions.com/Webster/www.artofasm.com



- ▶ Java VM für diverse Programmierbeispiele
- ▶ HADES Schaltungssimulator — HAMBURG DEsign System
tams.informatik.uni-hamburg.de/applets/hades/webdemos
- ▶ Assembler und Tools, Debugger
- ▶ C-Compiler
- ▶ GNU-Toolchain empfehlenswert: gcc, binutils, gdb
- ▶ unter Windows
 - ▶ Cygwin mit Development-Tools installieren www.cygwin.com
 - ▶ Linux VM mit vorinstallierter Software (VirtualBox, VMware ...)
- ▶ Links und weitere Infos im Moodle