

# Systemprogrammierung Unix/Linux



TECHNISCHE HOCHSCHULE NÜRNBERG  
GEORG SIMON OHM

Sommersemester 2014

Hans-Georg Eßer

[h.g.esser@cs.fau.de](mailto:h.g.esser@cs.fau.de)  
<http://ohm.hgesser.de/>

# Zur Vorlesung (1)

## Termine Systemprogrammierung

Vorlesung: Di 15:45-17:15 Uhr HQ104  
Praktikum: Di 17:30-19:00 Uhr HQ204

## Web-Seite

<http://ohm.hgesser.de/>

(Folien, Videos, Termine,  
sonstige Informationen)

# Zur Vorlesung (2)

## Hilfreiche Vorkenntnisse:

- **C** – Grundlagen der Programmierung in C (oder C++, C#, Java)  
→ C-Crashkurs (im Praktikumsteil)
- Betriebssysteme (4. Sem.),  
Rechnersysteme (3. Sem.)
- **Unix-Shell** – Benutzung der Standard-Shell *bash* unter Linux → Bash-Crashkurs

# Zur Vorlesung (3)

	Nr. in SPO	Modulgruppe / Modul	SWS   Leistungspunkte im Studienplansemester								Prüfung Art u. Dauer in Min. *)	
			3.	4.	5.	6.	7.					
5	Theoretische Grundlagen											
	12	Mathematik III (Statistik)		4	5						SchrP 90	
	13	Kryptographie und Informationssicherheit	6	7							SchrP 90	
	14	Algorithmen u. Datenstrukturen		6	7						SchrP 90	
6	Softwaremethoden und -technologien											
	15	Interaktive Systeme	6	7							SchrP 90	
	16	Datenbanken		6	6						SchrP 90	
	17	Web-Programmierung		4	5							
	18	Softwarearchitektur					4	5			SchrP 90	
	19	Programmiersprachen					6	7			SchrP 90	
	20	IT-Anwendungen (Projektarbeit)					2	3	4	6	StA, Ref	
7	Systeminfrastrukturen											
	21	Betriebssysteme		6	7						SchrP 90	
	22	Rechnersysteme	6	7							SchrP 90	
	23	Rechnerkommunikation und -netze	6	7							SchrP 90	
8	24	Fachwissenschaftliche Wahlpflichtmodule	SysPro Unix/Linux → BS-Entwicklung?					12	15	8	10	fachspezifisch

# Zur Vorlesung (4)

## Termine im Sommersemester 2014

4 SWS, d. h.

- Vorlesung/Praktikum: 14 x 180 min.  
18.03., 25.03., 01.04., 08.04., 15.04., 29.04., 06.05.,  
13.05., 20.05., 27.05., 03.06., 17.06., 24.06., 01.07.2014  
(evtl.: auch 08.07. ⚡ Prüfungszeitraum)

## Service / Web-Seite

- Folien und Praktikumsaufgaben
- Vorlesungs-Videos

# Zur Vorlesung (5)

**Aufwand: 5 ECTS** (1 ECTS-Punkt  $\approx$  25–30 h)  
**5 x 25 = 125 Stunden, davon:**

- 56 Stunden Präsenz (Vorlesung + Praktikum)
- $\leq 20$  Stunden: Vertiefte Einarbeitung in C, insbes. Pointer
- $\leq 20$  Stunden: Informationen in Fachbüchern nachlesen
- $\leq 20$  Stunden: Vorbereitung des Projektvortrags, ggf. mit Probevortrag

# Hans-Georg Eßer

- Dipl.-Math. (RWTH Aachen, 1997)  
Dipl.-Inform. (RWTH Aachen, 2005)
- Chefredakteur einer Linux-Zeitschrift (seit 2000) und Autor diverser Linux-Bücher
- seit 2006 Lehraufträge an verschiedenen Hochschulen: Betriebssysteme, Rechnerarchitektur, Informatik-Grundlagen, Systemprogrammierung, Betriebssystem-Entwicklung
- Seit 2010 Doktorand an der Univ. Erlangen-Nürnberg – Thema: Design und Implementation des Lehrbetriebssystems Ulix (Literate Programming)
- Betreuung von Abschlussarbeiten (Implem. Betriebssysteme)

# Kontakt

**Sprechstunde: –**

**Fragen:**

- direkt in der Vorlesung
- oder danach
- oder per E-Mail ([h.g.esser@cs.fau.de](mailto:h.g.esser@cs.fau.de), bitte nicht die Ohm-Adresse...)



```
Sep 19 14:20:18 amd64 sshd[20494]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61557
Sep 19 14:27:41 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 01:00:01 amd64 /usr/sbin/cron[29278]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 20 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 02:00:01 amd64 /usr/sbin/cron[30103]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 20 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 12:46:44 amd64 sshd[6516]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62004
Sep 20 12:46:44 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 12:48:41 amd64 sshd[6609]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62105
Sep 20 12:54:44 amd64 sshd[6694]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62514
Sep 20 15:27:35 amd64 sshd[9077]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64242
Sep 20 15:27:35 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:37:11 amd64 sshd[10102]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63375
Sep 20 16:37:11 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:38:10 amd64 sshd[1140]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63546
Sep 21 01:00:01 amd64 /usr/sbin/cron[17055]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 21 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 02:00:01 amd64 /usr/sbin/cron[17878]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 21 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:43:26 amd64 sshd[1088]: Accepted rsa for esser from ::ffff:87.234.201.207 port 6397
Sep 21 17:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:53:39 amd64 sshd[31269]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 21 18:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 19:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 01:00:01 amd64 /usr/sbin/cron[4674]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 22 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 02:00:01 amd64 /usr/sbin/cron[5499]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 22 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 20:23:21 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 01:00:01 amd64 /usr/sbin/cron[24739]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 23 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 02:00:01 amd64 /usr/sbin/cron[25555]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 23 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 18:04:05 amd64 sshd[6554]: Accepted publickey for esser from ::ffff:192.168.1.5 port 59771 ssh2
Sep 23 18:04:05 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 18:04:34 amd64 sshd[6606]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62093
Sep 24 01:00:01 amd64 /usr/sbin/cron[12436]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 24 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 02:00:01 amd64 /usr/sbin/cron[13253]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 24 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 11:15:48 amd64 sshd[20998]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64456
Sep 24 11:15:48 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 13:49:08 amd64 sshd[23197]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61330
Sep 24 13:49:08 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_seq_midi_event: unsupported module, tainting kernel.
Sep 24 15:42:07 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_seq_oss: unsupported module, tainting kernel.
Sep 24 20:25:31 amd64 sshd[29399]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62566
Sep 24 20:25:31 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 01:00:02 amd64 /usr/sbin/cron[662]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 25 01:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 02:00:01 amd64 /usr/sbin/cron[1484]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 25 02:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:25 amd64 sshd[8889]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64183
Sep 25 10:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:47 amd64 sshd[8921]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64253
Sep 25 11:30:02 amd64 sshd[9372]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62029
Sep 25 11:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:05:37 amd64 sshd[11554]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62822
Sep 25 14:05:37 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:06:10 amd64 sshd[11586]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62951
Sep 25 14:07:17 amd64 sshd[11608]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63392
Sep 25 14:08:33 amd64 sshd[11630]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63709
Sep 25 15:25:33 amd64 sshd[12930]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62778
```

# 1. Einführung / Motivation

# Warum Systemprogrammierung?

- verstehen, wie Programme, Bibliotheken und Betriebssystem zusammen arbeiten
- Hauptspeicher effizient nutzen
- systemnahe Programmierung z. B. für Embedded-Systeme wichtig
- nützliche Ergänzung zu Vorlesung „Betriebssysteme“

# Warum Unix/Linux?

- lange Tradition (Unix: 1969; C: 1972)
- für alle Windows-Spezialisten eine nützliche Ergänzung (Linux im Aufwärtstrend ...)
- Welt des Web stark Unix/Linux-dominiert, performante Webseiten-Entwicklung ist auch systemnah

```
Sep 19 14:20:18 amd64 sshd[20494]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61557
Sep 19 14:27:41 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 01:00:01 amd64 /usr/sbin/cron[29278]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 20 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 02:00:01 amd64 /usr/sbin/cron[30103]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 20 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 12:46:44 amd64 sshd[6516]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62004
Sep 20 12:46:44 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 12:48:41 amd64 sshd[6609]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62105
Sep 20 12:54:44 amd64 sshd[6694]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62514
Sep 20 15:27:35 amd64 sshd[9077]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64242
Sep 20 15:27:35 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:37:11 amd64 sshd[10102]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63375
Sep 20 16:37:11 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:38:10 amd64 sshd[10140]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63546
Sep 21 01:00:01 amd64 /usr/sbin/cron[17055]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 21 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 02:00:01 amd64 /usr/sbin/cron[17878]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 21 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:43:26 amd64 sshd[31088]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63397
Sep 21 17:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:53:39 amd64 sshd[31269]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 21 18:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 19:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 01:00:01 amd64 /usr/sbin/cron[4674]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 22 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 02:00:01 amd64 /usr/sbin/cron[5499]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 22 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 20:23:21 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 01:00:01 amd64 /usr/sbin/cron[24739]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 23 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 02:00:01 amd64 /usr/sbin/cron[25555]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 23 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 18:04:05 amd64 sshd[6554]: Accepted pubkey for esser from ::ffff:92.168.1.3 port 59771
Sep 23 18:04:05 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 18:04:34 amd64 sshd[6606]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62093
Sep 24 01:00:01 amd64 /usr/sbin/cron[12436]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 24 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 02:00:01 amd64 /usr/sbin/cron[13253]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 24 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 11:15:48 amd64 sshd[20998]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64456
Sep 24 11:15:48 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 13:49:08 amd64 sshd[23197]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61330
Sep 24 13:49:08 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_seq_midi_event: unsupported module, tainting kernel.
Sep 24 15:42:07 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_seq_oss: unsupported module, tainting kernel.
Sep 24 20:25:31 amd64 sshd[29399]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62566
Sep 24 20:25:31 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 01:00:02 amd64 /usr/sbin/cron[662]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 25 01:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 02:00:01 amd64 /usr/sbin/cron[1484]: (root) CMD (/sbin/evlogmgr -c 'age > "30d"')
Sep 25 02:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:25 amd64 sshd[8889]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64183
Sep 25 10:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:47 amd64 sshd[8921]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64253
Sep 25 11:30:02 amd64 sshd[9372]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62029
Sep 25 11:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:05:37 amd64 sshd[11554]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62822
Sep 25 14:05:37 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:06:10 amd64 sshd[11586]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62951
Sep 25 14:07:17 amd64 sshd[11608]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63392
Sep 25 14:08:33 amd64 sshd[11630]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63709
Sep 25 15:25:33 amd64 sshd[12930]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62778
```

# Gliederung

# Gliederung (1)

1. Einleitung
2. Einführung in C und die Linux-Shell Bash
3. Software und Betriebssystem; System Calls
4. Prozesse: fork, exec, wait
5. Dateien: open, read, write, close
6. Fortgeschrittene I/O
7. Speicherverwaltung: malloc, free, memcpy
8. Nebenläufigkeit mit POSIX-Threads
9. Einführung ins Projekt

# Gliederung (2)

- Zu einigen Themen erst Grundlagen (Theorie der Betriebssysteme)
- Vorstellung der Features, die Unix/Linux hier bietet
- Programmieraufgaben dazu

Gegen Ende des Semesters: größeres Programmierprojekt

# 2. Crashkurse C und Bash

## Gliederung

1. Einleitung
2. C und Bash
3. Software/BS,  
System Calls
4. Prozesse
5. Dateien
6. Fortgeschr. I/O
7. Speicher
8. Threads

- *nicht* objektorientierte Programmierung
- Variablen
- Funktionen
- C-Structs
- Pointer und Arrays
- Linux-Shell Bash: Crashkurs

# 3. Software / BS / System Calls

## Gliederung

1. Einleitung
2. C und Bash
3. Software/BS,  
System Calls
4. Prozesse
5. Dateien
6. Fortgeschr. I/O
7. Speicher
8. Threads

- System Calls (Syscalls), Beispiele
- Syscall in Assembler aufrufen
- Syscall in C aufrufen
- Bibliotheksfunktionen
- Beispiel: Dateizugriff mit `fread()`



# 4. Prozesse

## Gliederung

1. Einleitung
2. C und Bash
3. Software/BS,  
System Calls
4. Prozesse
5. Dateien
6. Fortgeschr. I/O
7. Speicher
8. Threads

- Prozesskonzept unter Unix/Linux
  - Prozesskontrollblock
  - PID, Vater/Sohn, Baumstruktur
- Prozessverwaltung in der Shell
- Neue Prozesse erzeugen (fork)
- Programm in Prozess laden (exec)
- Warten auf Prozess (wait)
- Signalisierung (kill, signal),  
Signal-Handler

# 5. Dateien

## Gliederung

1. Einleitung
2. C und Bash
3. Software/BS,  
System Calls
4. Prozesse
5. Dateien
6. Fortgeschr. I/O
7. Speicher
8. Threads

- Dateisysteme unter Unix/Linux
  - Datei, Inode, Verzeichnis
  - Link, Symlink, Named Pipe
- Datei öffnen und schließen, file descriptor
- lesen, schreiben, Position
- fork und offene Dateien
- Low-Level- und Bibliotheksfunktionen (open, read, ... vs. fopen, fread, ...)

# 6. Fortgeschrittene I/O

## Gliederung

1. Einleitung
2. C und Bash
3. Software/BS,  
System Calls
4. Prozesse
5. Dateien
6. Fortgeschr. I/O
7. Speicher
8. Threads

- Standard-Eingabe, -Ausgabe und -Fehlerausgabe, inkl. file descriptors
- Bibliotheksfunktionen printf und scanf
- Pipes, dup()
- I/O-Multiplexing mit select()
- Memory-mapped files

# 7. Speicher

## Gliederung

1. Einleitung
2. C und Bash
3. Software/BS,  
System Calls
4. Prozesse
5. Dateien
6. Fortgeschr. I/O
7. Speicher
8. Threads

- Speicherverwaltung in C
- Organisation des Prozess-Speichers (Code, Daten, Stack, Heap)
- Speicher reservieren (malloc)
- Speicher freigeben (free)

# 8. Threads

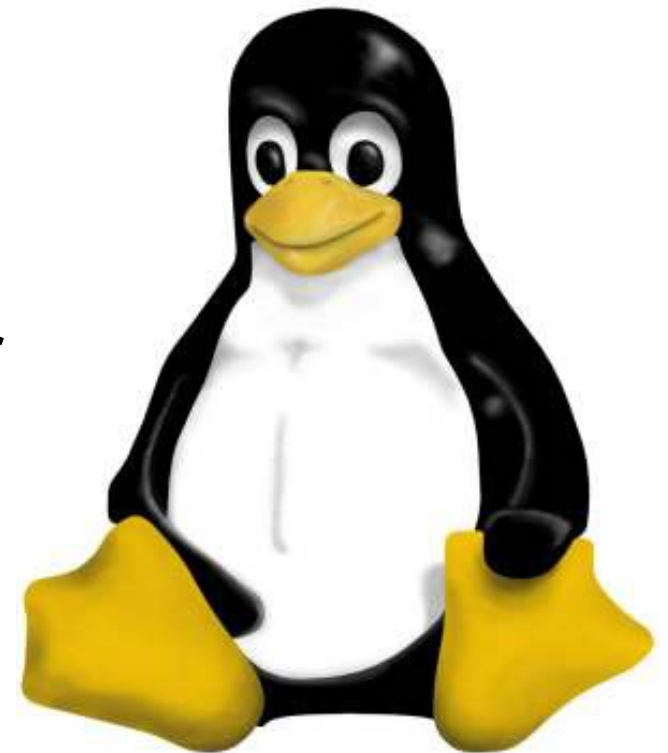
## Gliederung

1. Einleitung
2. C und Bash
3. Software/BS,  
System Calls
4. Prozesse
5. Dateien
6. Fortgeschr. I/O
7. Speicher
8. Threads

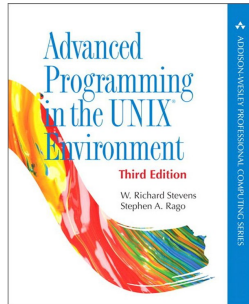
- Nebenläufigkeit innerhalb einer Anwendung
- Speichermodell Threads
- POSIX-Threads
  - erzeugen (`pthread_create`)
  - warten (`pthread_join`)
- Vergleich Threads / Prozesse

# Linux

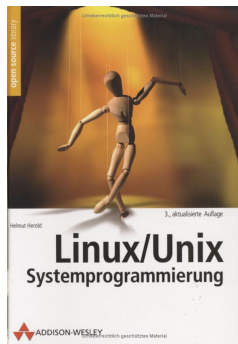
- Offene Kernel-Quellen:
  - nachlesen, wie etwas geht
  - ändern, was nicht gefällt
- Etabliertes Standardsystem für sehr viele Plattformen (PC Desktop / Server, Embedded etc.)
- läuft auch im Praktikumsraum



# Literatur



**Advanced Programming in the UNIX Environment**  
W. Richard Stevens, Stephen A. Rago  
2013 (3rd edition)



**Linux/Unix Systemprogrammierung**  
Helmut Herold (TH Nürnberg)  
2004 (3. Auflage)

**A Tutorial on Pointers and Arrays in C**  
Ted Jensen, 50 S., 2003  
<http://home.earthlink.net/~momotuk/pointers.pdf>