

Betriebssysteme und nebenläufige Programmierung

SoSe 2026

Roland Wismüller
Betriebssysteme / verteilte Systeme
roland.wismueller@uni-siegen.de
Tel.: 0271/740-4050, Büro: H-B 8404

Stand: 20. März 2026

Betriebssysteme und nebenläufige Programmierung

SoSe 2026

12 Zusammenfassung, wichtige Themen



➔ Prozesse und Threads

➔ Zustandsgraph

➔ Elemente des Prozeß- bzw. Thread-Kontrollblocks

➔ Ablauf von Interrupt, Ausnahme, Systemaufruf

➔ Threadwechsel

➔ Synchronisation

➔ Kritischer Abschnitt, wechselseitiger Ausschluß

➔ (Lösungen mit Schreib-/Leseoperationen, *Spin locks*)

➔ Semaphore

➔ Monitore

➔ **Java *Locks* und Bedingungsvariable** (nur 6 LP)

➔ [Code angeben bzw. gegebener Code mit Fragen]



- ➔ (Kommunikation)
- ➔ **Verklemmungen**
 - ➔ Definition und Bedingungen
 - ➔ *Deadlock*-Erkennung, v.a.: **Algorithmen**
 - ➔ *Deadlock-Avoidance*, v.a.: **sichere Zustände, Bankiers-Alg.**
 - ➔ *Deadlock-Prevention*
- ➔ **Scheduling**
 - ➔ präemptiv, nicht-präemptiv
 - ➔ **FCFS, SJF, RR, Prioritäten, Multilevel-Scheduling**



➔ **Speicherverwaltung**

- ➔ logischer / physischer Adreßraum
- ➔ Speicherschutz
- ➔ Zuteilung zusammenhängender Speicherbereiche
 - ➔ Swapping, Relokation, (dynamische Speicherverwaltung)

➔ **Paging**

- ➔ **Prinzip**; Seiten, Kacheln, Seitentabelle
- ➔ **Ablauf der Umsetzung**
 - ➔ ein- und zweistufige Seitentabelle
- ➔ virtueller Speicher
 - ➔ Grundlagen: Lokalität, *Working Set*, *Resident Set*
 - ➔ **Ablauf eines Seitenwechsels**
 - ➔ **Seitenersetzungsalgorithmen**: Belady, **NRU**, **FIFO**, ***Second Chance*** / ***Clock***, **LRU**



- ➔ Ein-/Ausgabe und Dateisysteme
 - ➔ Programmierte E/A, Interrupt-gesteuerte E/A, DMA
 - ➔ (Schichten der E/A-Software)
 - ➔ (Festplatten: Aufbau, Zugriffszeit)
 - ➔ (Schichten des Dateisystems)
 - ➔ Zuteilung von Blöcken an Dateien: verteilte Belegung
- ➔ Schutz
 - ➔ Schutzmatrix, ACL, Capability

- ➔ 1.6 Systemaufrufe
- ➔ 2.2 Nebenläufige Programmierung
- ➔ 3.5 Synchronisation in Mehrprozessorsystemen
- ➔ 3.6 Speicherkonsistenz
- ➔ 3.11 Lock-free Datenstrukturen
- ➔ 3.12 Transactional Memory
- ➔ 6 Koroutinen und asynchrone Programmierung
- ➔ 7.5 Scheduling: Beispiele
- ➔ 9.5 Moderne Dateisysteme
- ➔ 11 Virtualisierung



- ➔ Programmierung im Detail
 - ➔ Shell-Programmierung
 - ➔ Thread-Programmierung mit C, C++, Linux
 - ➔ Abschnitte 2.7.2 - 2.7.4 und 3.10.3 - 3.10.5
 - ➔ konkrete Java-Programmierung
 - ➔ **der Umgang mit *Locks* und Bedingungsvariablen ist aber Klausurstoff (für 6 LP)!**

- ➔ Termine: **Do., 28.08.2025, 12:00 Uhr (s.t.), Raum AR-E 8101 (Audimax)**
 - ➔ nächster Termin nach dem WiSe 25/26
- ➔ Klausurdauer: **60 min, ohne Hilfsmittel**
- ➔ Anmeldung: **bis 14 Tage vorher**