



# Module Description 39-Inf-17\_a Operating Systems

Faculty of Technology

*Version dated Feb 11, 2026*

This module guide reflects the current state and is subject to change. Up-to-date information and the latest version of this document can be found online via the page

<https://ekvv.uni-bielefeld.de/sinfo/publ/modul/79251402>

The current and valid provisions in the module guide are binding and further specify the subject-related regulations (German "FsB") published in the Official Announcements of Bielefeld University.

## **39-Inf-17\_a Operating Systems**

---

### **Faculty**

---

Faculty of Technology

### **Person responsible for module**

---

Prof. Dr.-Ing. Ralf Möller

### **Regular cycle (beginning)**

---

Discontinued

### **Credit points**

---

5 Credit points

### **Competencies**

---

Die Veranstaltung Betriebssysteme vermittelt grundlegende Betriebssystem-Konzepte am Beispiel des Betriebssystems Unix/Linux. In den Übungen werden die Konzepte anhand kleiner Projekte zur betriebssystemnahen Programmierung in der Programmiersprache C vertieft. Die Studierenden sollen in die Lage versetzt werden, Betriebssystem-Konzepte zu verstehen und eigenständig betriebssystemnahe Programme zu schreiben, wobei der Schwerpunkt auf der Erzeugung und Steuerung von Prozessen, dem Entwurf threadbasierter Programme und der Anwendung von Synchronisationsmechanismen (z.B. Mutex-Locks, Semaphoren) in Zusammenhang mit Threads liegt.

### **Content of teaching**

---

Die Vorlesung Betriebssysteme vermittelt grundlegende Betriebssystem-Konzepte: Aufgaben und Architektur von Betriebssystemen, Prozess- und Thread-Verwaltung, Synchronisationsmechanismen, Speicherverwaltung, Ein-Ausgabe- und Datei-Verwaltung. Die Konzepte werden am Beispiel des Betriebssystems Unix und des Linux-Kernels veranschaulicht. In den Übungen werden eine Reihe kleinerer Projekte zur betriebssystemnahen Programmierung in C bearbeitet (z.B. einfache Shell, threadbasierter Server, Scheduler). In der Vorlesung Betriebssysteme wird begleitend die Programmiersprache C behandelt. Dies erleichtert den Einstieg in die Programmierübungen.

### **Recommended previous knowledge**

---

Rechnerarchitektur (39-Inf-8)

### **Necessary requirements**

---

—

### **Explanation regarding the elements of the module**

---

Notwendigkeit von zwei Modulteilprüfungen: In der Klausur werden vorwiegend theoretische Kenntnisse geprüft. In den Übungen werden vorwiegend anwendungsorientierte Kompetenzen in Form eines Portfolios geprüft.

Module structure: 2 uPr<sup>1</sup>

## Courses

Title	Type	Regular cycle	Workload 5	Lp <sup>2</sup>
<b>Betriebssysteme</b>	exercise	SoSe	60 h (30 + 30)	2 [Pr]
<b>Betriebssysteme</b>	lecture	SoSe	90 h (60 + 30)	3 [Pr]

## Examinations

Allocated examiner	Type	Weighting	Workload	Lp <sup>2</sup>
<p>Teaching staff of the course <b>Betriebssysteme (exercise)</b></p> <p><i>Portfolio aus Übungsaufgaben und Aufgaben, die auf Programmieraufgaben vorbereiten, (Bestehensgrenze 50% der erzielbaren Punkte) und Programmieraufgaben (Bestehensgrenze 50% der erzielbaren Punkte), die jeweils veranstaltungsbezogen gestellt werden.</i></p> <p><i>Die Kontrolle der Übungsaufgaben und Programmieraufgaben umfasst auch direkte Fragen zu den Lösungsansätzen, die von den Studierenden in den Übungen beantwortet werden müssen. Die Veranstalterin/der Veranstalter kann ein individuelles Erläutern und Vorführen von Aufgaben verlangen sowie einen Teil der Übungs- bzw. Programmieraufgaben durch Präsenzübungen ersetzen.</i></p> <p><i>Die Aufgaben im Rahmen des Portfolios werden in der Regel wöchentlich ausgegeben.</i></p>	Portfolio	without grades	-	-
<p>Teaching staff of the course <b>Betriebssysteme (lecture)</b></p> <p><i>Die Klausur (60-90 min.) bezieht sich auf den Stoff der Vorlesung und der Übung. Aus wichtigem Grund im Ausnahmefall kann ein Essay (4 Seiten) mit einer stark auf die vermittelten Kenntnisse und Fähigkeiten bezogenen Aufgabenstellung ODER ein Essay (bis zu 4 A4-Seiten als Abschlussbericht) mit einer stark auf die vermittelten Kenntnisse und Fähigkeiten bezogenen Programmieraufgabe von der*dem Lehrenden vorgesehen werden.</i></p> <p><i>Durch die Auseinandersetzung mit der Aufgabenstellung (bzw. Programmieraufgabe) weisen die Studierenden exemplarisch Kenntnisse und Fähigkeiten nach, wobei es erforderlich ist, den gesamten Kontext des Moduls zu berücksichtigen.</i></p>	Essay o. Klausur	without grades	-	-

## Further notices

---

Bei diesem Modul handelt es sich um ein eingestelltes Angebot. Ein entsprechendes Angebot, um dieses Modul abzuschließen, wurde bis maximal Wintersemester 2023/24 vorgehalten.  
Bisheriger Angebotsturnus war jedes Sommersemester.

## Legend

---

- 1 The module structure displays the required number of study requirements and examinations.
- 2 LP is the short form for credit points.
- 3 The figures in this column are the specialist semesters in which it is recommended to start the module. Depending on the individual study schedule, entirely different courses of study are possible and advisable.
- 4 Explanations on mandatory option: "Obligation" means: This module is mandatory for the course of the studies; "Optional obligation" means: This module belongs to a number of modules available for selection under certain circumstances. This is more precisely regulated by the "Subject-related regulations" (see navigation).
- 5 Workload (contact time + self-study)

**SoSe** Summer semester

**WiSe** Winter semester

**SL** study requirement

**Pr** Examination

**bPr** Number of examinations with grades

**uPr** Number of examinations without grades