

LEHRVERANSTALTUNGSBESCHREIBUNG

Betriebssysteme

Akademisches Jahr **2025-2026**

1. Angaben zum Programm

1.1. Hochschuleinrichtung	Babeş-Bolyai Universität
1.2. Fakultät	Mathematik und Informatik
1.3. Department	Informatik
1.4. Fachgebiet	Informatik
1.5. Studienform	Bachelor
1.6. Studiengang / Qualifikation	Informatik
1.7. Form des Studiums	Vollzeit

2. Angaben zum Studienfach

2.1. LV-Bezeichnung	Betriebssysteme	Code der LV	MLG5007
2.2. Lehrverantwortlicher – Vorlesung	Dozent Dr. Sanda-Maria AVRAM		
2.3. Lehrverantwortlicher – Seminar	Dozent Dr. Sanda-Maria AVRAM		
2.4. Studienjahr	1	2.5. Semester	2
		2.6. Prüfungsform	E
		2.7. Art der LV	Pflichtfach

3. Geschätzter Workload in Stunden

3.1. SWS	5	von denen: 3.2 Vorlesung	2	3.3. Seminar/Übung/Project	3
3.4. Gesamte Stundenanzahl im Lehrplan	70	von denen: 3.5 Vorlesung	28	3.6 Seminar/Übung/Project	42
Verteilung der Studienzeit:					Std.
Studium nach Handbücher, Kursbuch, Bibliographie und Mitschriften					17
Zusätzliche Vorbereitung in der Bibliothek, auf elektronischen Fachplattformen und durch Feldforschung					7
Vorbereitung von Seminaren/Übungen, Präsentationen, Referate, Portfolios und Essays					15
Tutorien					3
Prüfungen					10
Andere Tätigkeiten:					3
3.7. Gesamtstundenanzahl Selbststudium		55			
3.8. Gesamtstundenanzahl / Semester		125			
3.9. Anrechnungspunkte		5			

4. Voraussetzungen (falls zutreffend)

4.1. zur Lehrveranstaltung	
4.2. kompetenzbezogene	<ul style="list-style-type: none"> • Minimale Programmierkenntnisse in C-Standard.

5. Bedingungen (falls zutreffend)

5.1. zur Durchführung der Vorlesung	<ul style="list-style-type: none"> • Vorlesungsraum mit Beamer
5.2. zur Durchführung des Seminars / der Übung	<ul style="list-style-type: none"> • Labor mit Computern mit Internetanschluss und UNIX-Betriebssystem oder Zugang zu einem UNIX-Server

6.1. Spezifische erworbene Kompetenzen¹

Berufliche/ Wesentliche Kompetenzen	<ul style="list-style-type: none"> • unter Verwendung der theoretischen Grundlagen der Informatik und der formalen Modellierung • Entwicklung und Wartung von IT-Anwendungen
Transversale Kompetenzen	<ul style="list-style-type: none"> • Anwendung der Regeln der organisierten und effizienten Arbeit, verantwortungsbewusste Haltung gegenüber dem didaktisch-wissenschaftlichen Bereich, um das eigene Potenzial kreativ zu nutzen, unter Beachtung der Grundsätze und Regeln der Berufsethik

6.2. Lernergebnisse

Kennt-nisse	<ul style="list-style-type: none"> • Der Absolvent verfügt über Grundkenntnisse der betriebssystemspezifischen Programmierung sowie allgemeine Programmiergrundlagen • Der Absolvent kennt die Konzepte der Softwaremodellierung und kann funktionale sowie nicht-funktionale Anforderungen, wie in spezifischen Dokumenten beschrieben, umsetzen • Der Absolvent beherrscht Methoden zur Testung und Verifikation von Softwaresystemen
Fähigkeiten	<ul style="list-style-type: none"> • Der Absolvent verfügt über die erforderlichen Fähigkeiten, Betriebssysteme zu installieren und zu konfigurieren • Der Absolvent kennt die Werkzeuge zur Testung, Fehlerbehebung (Debugging) und Validierung von Softwareanwendungen
Verantwortung und Autonomie	<ul style="list-style-type: none"> • Der Absolvent ist in der Lage, komplexe Problemstellungen zu identifizieren und zusammenhängende Fragestellungen zu untersuchen, um Lösungen zu entwerfen und umzusetzen • Der Absolvent verfügt über die notwendigen Fähigkeiten, um Computerprogramme zu entwerfen und Softwaresysteme zu analysieren.

7. Ziele (entsprechend der erworbenen Kompetenzen)

7.1 Allgemeine Ziele der Lehrveranstaltung	<ul style="list-style-type: none"> • Assimilation der wichtigsten Konzepte von Betriebssystemen
---	--

¹ Man kann Kompetenzen oder Lernergebnisse, oder beides wählen. Wenn nur eine Option ausgewählt wird, wird die Tabelle für die andere Option gelöscht, und die beibehaltene Option erhält die Nummer 6.

7.2 Spezifische Ziele der Lehrveranstaltung	<ul style="list-style-type: none"> • Erlernen der wichtigsten Funktionen des Betriebssystems Unix • Shell Programmierkenntnisse und Textdateiverarbeitung unter Unix • Verwaltung von Multitasking-Anwendungen mithilfe von Unix-Prozessen
--	---

8. Inhalt

8.1 Vorlesung	Lehr-und Lernmethode	Anmerkungen
1-3 Unix OS: Externe Schnittstellen <ul style="list-style-type: none"> • Allgemeine Struktur des Betriebssystems • Reguläre Ausdrücke, Dateispezifikation, generische Spezifikation <ul style="list-style-type: none"> • Filter; allgemeine Grundsätze sort, awk, sed, sed, grep • sh, csh, ksh, bash; allgemeine Einführung • Nützliche Shell-Befehle und externe Prozessverwaltung • Shell-Programmierung; Shell-Anwendungen • Die Struktur von Verzeichnissen im Unix-System • Mounting-Konzept • Symbolische und harte Links 	Darstellung der Thematik, Diskussion, Befragung, Entdeckung	
4-7 Unix-Betriebssystem: Systemaufrufe, interne Strukturen <ul style="list-style-type: none"> • Dateien und Prozesse unter Unix • E/A in C POSIX: open, close, lseek, read, write, dup, dup2 • Dateischutz • Prozesse unter Unix; Struktur eines Prozesses • Systemaufrufe zur Prozessverwaltung: fork, wait, exit, exec* • Kommunikation zwischen Prozessen: pipe, popen, FIFO • POSIX-Threads 	Darstellung der Thematik, Diskussion, Befragung, Entdeckung	
8-9 Dateisysteme <ul style="list-style-type: none"> • Allgemeine Festplattenverwaltung und Dateisysteme • Planung von Magnetplattenzugriffen • Interne Struktur von DOS-Platten und -Dateisystemen; FAT-Tabelle • WindowsNT & 2000: interne Struktur des Festplatten- und Dateisystems; NTFS-Mechanismus, MFT-Datei • Interne Struktur von Unix-Platten und -Dateisystemen; i-node-Mechanismus 	Darstellung der Thematik, Diskussion, Befragung, Entdeckung	
10-14 Allgemeine Theorie der Betriebssysteme <ul style="list-style-type: none"> • Arten von Computersystemen und Betriebssystemen. • E/A-Kanal, mehrere Puffer. Multiprogramming. • Allgemeiner Aufbau und Funktionen eines Betriebssystems • Prozesse: Spezifikation, Gleichzeitigkeit, Semaphoren, Deadlock • Ablaufplanung von Prozessen • Speicherverwaltung • Swapplanung zwischen internem und sekundärem Speicher 	Darstellung der Thematik, Diskussion, Befragung, Entdeckung	

Literatur

Auf deutscher Sprache:

1. **Bartelmann, K.**, Betriebssysteme, Uni Wien, 2005.
2. **Gräfe, M.**, Die Möglichkeiten des Betriebssysteme mit eigenen Programmen nutzen, Carl Hansen Verlag, München, 2010.
3. **Herold, H., Arndt, J.**, C-Programmierung: Unter Linux, Unix und Windows, 2013.
4. **Jäger, M.**, Betriebssysteme, FH Giessen, 2004.
5. **Kofler, M.**, Linux Kommandoreferenz: Shell-Befehle von A bis Z, Rheinwerk Verlag, Bonn, 2016.
6. **Küchlin, W.**, Betriebssysteme, Uni Wien, 2004.
7. **Küchlin, W., Weber, A.**, Einführung in die Informatik, Springer, Berlin, Heidelberg, New York, 2004.
8. **Glatz, E.**, Betriebssysteme: Grundlagen, Konzepte, Systemprogrammierung, dpunkt.verlag, Heidelberg, 2015.
9. **Tanenbaum, A.**, Moderne Betriebssysteme, Pearson Studium, München, 2009.

Auf englischer Sprache:

10. **Albing, C., Vossen, J.P., Newhman, C.**, bash Cookbook: Solutions and Examples for bash Users, O'Reilly, USA, 2007.
11. **Kernighan, B.W., Dennis, R.M.**, The C Programming Language, Prentice Hall, Massachusetts, 2012.
12. **Stallings, W.**, Operating Systems: Internals and Desing Principles, Pearson Education Limited, Essex, 2015.
13. **Raymond, E.S.**, The Art of UNIX Programming, Addison-Wesley, Pearson Education Limited, USA, 2004.
14. **Tanenbaum, A., Herbert, B.**, Modern Operating Systems, Pearson Education Limited, Essex, 2015.

Auf rumänischer Sprache:

15. **Boian, F., Vancea, A., Boian, R., Bufnea, D., Sterca, A., Cobarzan, C., Cojocar, D.**, Sisteme de operare, Ed. Risoprint, Cluj-Napoca, 2006.

8.2 Seminar	Lehr-und Lernmethode	Anmerkungen
1. UNIX: Befehle und Texteditoren	Beispiele, Diskussionen	
2. <i>sed</i> und <i>grep</i>		
3. <i>awk</i>		
4. UNIX-Prozesse		
5. Interprozesskommunikation: <i>pipe</i>		
6. Interprozesskommunikation: <i>FIFO</i>		
7. Wiederholung		
8.2 Laborarbeit	Lehr-und Lernmethode	Anmerkungen
1-2. UNIX: Befehle und Texteditoren	Beschreibung, Erklärung, Debatte, Beispiele	
3. shell 1		
4. <i>sed</i> und <i>grep</i>		
5. <i>awk</i>		
6. shell 2		
7-8. C Programmierung		
9. UNIX Prozesse		
10. Interprozesskommunikation: <i>pipe</i>		

11. Interprozesskommunikation: FIFO		
12. Unix-Threads		
13. Die Studierenden liefern die letzten Laboraufgaben. Vorbereitung der Abschlussprüfung		
14. Praktische Prüfung	Prüfung	
Literatur		
Auf deutscher Sprache:		
1. Gräfe, M. , Die Möglichkeiten des Betriebssysteme mit eigenen Programmen nutzen, Carl Hansen Verlag, München, 2010.		
2. Herold, H., Arndt, J. , C-Programmierung: Unter Linux, Unix und Windows, 2013.		
3. Kofler, M. , Linux Kommandoreferenz: Shell-Befehle von A bis Z, Rheinwer Verlag, Bonn, 2016.		
Auf englischer Sprache:		
4. Albing, C., Vossen, J.P., Newhman, C. , bash Cookbook: Solutions and Examples for bash Users, O'Reilly, USA, 2007.		
5. Kernighan, B.W., Dennis, R.M. , The C Programming Language, Prentice Hall, Massachusetts, 2012.		
6. Raymond, E.S. , The Art of UNIX Programming, Addison-Wesley, Pearson Education Limited, USA, 2004.		
Auf rumänischer Sprache:		
7. Boian, F., Vancea, A., Boian, R., Bufnea, D., Sterca, A., Cobarzan, C., Cojocar, D. , Sisteme de operare, Ed. Risoprint, Cluj-Napoca, 2006.		

9. Verbindung der Inhalte mit den Erwartungen der Wissensgemeinschaft, der Berufsverbände und der für den Fachbereich repräsentativen Arbeitgeber

- Diese Vorlesung wird an international bekannten Universitäten im Fachgebiet Informatik angeboten.
- Der Inhalt der Vorlesung entspricht der IEEE und ACM Richtlinien

10. Prüfungsform

Veranstaltungart	10.1 Evaluationskriterien	10.2 Evaluationsmethoden	10.3 Anteil an der Note (%)
10.4 Vorlesung	Kenntnis der Grundprinzipien des Fachgebiets	Abschlussprüfung (reguläre Sitzung)	40%
10.5 Seminar / Übung	Anwendung dieser Konzepte bei der Lösung von Problemen	Aufgaben im Labor (während des Semesters)	20%
	Entwicklung von Shells und Erstellung von Unix-Prozessen	Praktische Prüfung (letzte Semesterwoche)	40%

10.6 Minimale Leistungsstandards

- Mindestens die Note 5 (auf einer Skala von 1 bis 10) für alle Prüfungsarten.
- Seminarteilnahme von mindestens 75% (mindestens 5 von 7 Seminaren)
- Anwesenheit in den Übungen von mindestens 90% (mindestens 12 von 14 Übungen)
- Kenntnis der theoretischen und praktischen Aspekte von **Shell**-Konzepten und **Prozessen**:
 - **Shell**: Arbeiten mit Dateien, Kontrollstrukturen (insbesondere *for*), Zugriff auf Kommandozeilenparameter;
 - **Prozesse**: Einwegkommunikation über Pipe oder FIFO.

11. SDD-Nachhaltigkeits-Logos (Sustainable Development Goals)²

Nicht anwendbar.

Ausgefüllt am:	Vorlesungsverantwortlicherin	Seminarverantwortlicherin
15.04.2025	Dozent Dr. Sanda-Maria AVRAM	Dozent Dr. Sanda-Maria AVRAM
Genehmigt im Department am:		Departmentleiter
...		Dozent Dr. Adrian STERCA

² Bitte belassen Sie nur die Logos, die entsprechend den [Regularien zu Anwendung der Nachhaltigkeits-Logos im akademischen Betrieb](#) dem jeweiligen Studienfach entsprechen und löschen Sie diejenigen Logos, inklusive das allgemeine *Nachhaltigkeits-Logo* falls dieses nicht zutrifft. Falls keines der Logos für das Studienfach anwendbar ist, löschen Sie alle mit der Angabe „*nicht anwendbar*“.