

## Modul

# BVS1 - Betriebssysteme und Verteilte Systeme 1

Bachelor Technische Informatik 2020

Version: 1 | Letzte Änderung: 02.08.2019 11:33 | Entwurf: 0 | Status: vom Modulverantwortlichen freigegeben | Verantwortlich: Vogt

## ^ Allgemeine Informationen

<b>Anerkannte Lehrveranstaltungen</b>	<a href="#">BVS1_Vogt</a>
<b>Dauer</b>	1 Semester
<b>ECTS</b>	5
<b>Zeugnistext (de)</b>	Betriebssysteme und Verteilte Systeme 1
<b>Zeugnistext (en)</b>	Operating Systems and Distributed Systems 1
<b>Unterrichtssprache</b>	deutsch oder englisch
<b>abschließende Modulprüfung</b>	Ja

## Modulprüfung

<b>Benotet</b>	Ja
<b>Frequenz</b>	Jedes Semester

## Prüfungskonzept

Die Studierenden sollen in einer schriftlichen Klausur folgende Fähigkeiten nachweisen, die die o.a. Kompetenzen K1-K4 und K6-K9 abdecken: 1.) Sicherer Umgang mit grundlegenden Begrifflichkeiten, Konzepten und Techniken, 2.) Anwendung programmiersprachlicher und abstrakterer Konstrukte zur Lösung von Anwendungsproblemen bei der nebenläufigen und verteilten Programmierung, 3.) Prüfung von Lösungsvorschlägen auf Korrektheit. Typische Aufgabenformen zu 1.) sind Multiple-Choice-Fragen, Lückentexte, Bewertung von Aussagen hinsichtlich ihrer Korrektheit, zu 2.) Lösung kleinerer umgangssprachlich formulierter Probleme durch Programmstücke oder in abstrakterer Form und zu 3.) das Finden von Fehlern in

Aussagen und vorgegebenen Programmstücken.

Die übrigen Kompetenzen (K10, K13, K15, K16) lassen sich in einer schriftlichen Klausur begrenzter Dauer nur schwer sinnvoll prüfen. Sie werden daher im Praktikum geprüft, das als studienbegleitende Vorleistung absolviert werden muss.

## ^ Allgemeine Informationen

### Inhaltliche Voraussetzungen

<b>PI1 - Praktische Informatik 1</b>	Sicherer Umgang mit einer Programmiersprache.
<b>PI2 - Praktische Informatik 2</b>	Sicherer Umgang mit einer Programmiersprache.
<b>DR - Digitalrechner</b>	Kenntnisse über Aufbau und Funktionalität eines Digitalrechners.
<b>GSP - Grundlagen der Systemprogrammierung</b>	Grundkenntnisse über die hardwarenahe Programmierung eines Digitalrechners.
<b>NP - Netze und Protokolle</b>	Grundkenntnisse über Internet-Protokolle.

### Kompetenzen

<b>Kompetenz</b>	<b>Ausprägung</b>
Typische Werkzeuge, Standards und Best Practices der industriellen Praxis kennen und einsetzen	diese Kompetenz wird vermittelt
In Systemen denken	diese Kompetenz wird vermittelt
fachliche Probleme abstrahieren und formalisieren	diese Kompetenz wird vermittelt
Konzepte und Methoden der Informatik, Mathematik und Technik kennen und anwenden	diese Kompetenz wird vermittelt
In vorhandene Systeme einarbeiten und vorhandene Komponenten sinnvoll nutzen	diese Kompetenz wird vermittelt
Systeme analysieren	diese Kompetenz wird vermittelt
Systeme realisieren	diese Kompetenz wird vermittelt
Systeme prüfen	diese Kompetenz wird vermittelt

Informationen beschaffen und auswerten; Technische Zusammenhänge darstellen und erläutern diese Kompetenz wird vermittelt

---

Komplexe technische Aufgaben im Team bearbeiten diese Kompetenz wird vermittelt

---

Befähigung zum lebenslangen Lernen diese Kompetenz wird vermittelt

---

Kommunikative und interkulturelle Fähigkeiten anwenden diese Kompetenz wird vermittelt

## ^ Vorlesung / Übungen

### Exemplarische inhaltliche Operationalisierung

Die grundlegenden Komponenten und Aufgaben von Betriebssystemen und verteilten Systemen werden eingeführt und am Beispiel aktueller Systeme illustriert. Aufbauend darauf werden Programmierschnittstellen und Dienste zur Implementierung nebenläufiger Vorgänge betrachtet. Die Studierenden werden damit in die Lage versetzt, nebenläufig arbeitende, kommunizierende und kooperierende Programme zu verstehen und selbst zu erstellen.

### Separate Prüfung

keine

## ^ Praktikum

### Exemplarische inhaltliche Operationalisierung

Die Studierenden lernen, Konzepte und Techniken der nebenläufigen und verteilten Programmierung selbstständig zur Lösung von Anwendungsproblemen anzuwenden und dabei die Programmierschnittstellen von Systemsoftware zu nutzen. Dazu beschaffen sie auch selbstständig Informationen und passen vorhandene Software an.

### Separate Prüfung

<b>Benotet</b>	Nein
<b>Frequenz</b>	Einmal im Jahr
<b>Voraussetzung für Teilnahme an Modulprüfung</b>	Ja

Die Studierenden schließen sich zu Kleingruppen zusammen. Jede Kleingruppe absolviert mehrere "Praktikumsrunden" mit zugewiesenen Laborterminen. In jeder Runde werden Programmieraufgaben gelöst.

Zur Vorbereitung eines Labortermins muss ein "Vorbereitungsblatt" praktisch gelöst werden. Die dabei erworbenen Kenntnisse werden zu Beginn des Termins geprüft (kurzer schriftlicher Eingangstest, persönliches Gespräch mit dem Betreuer). Wird diese Prüfung nicht bestanden, so muss ein Folgetermin wahrgenommen werden; im Wiederholungsfall führt dies zum Nichtbestehen des Praktikums. Im Erfolgsfall wird ein "Laborarbeitsblatt" mit weiteren Aufgaben unter Aufsicht (und ggf. mit Hilfestellung) bearbeitet.

Durch diese stufenweise Vorgehensweise können sämtliche aufgeführte Kompetenzen individuell abgeprüft werden.