

# Modulhandbuch ARP

## Alternative Rechnerarchitekturen und Programmiersprachen

Master Technische Informatik 2020

Version: 1 | Letzte Änderung: 27.09.2019 11:24 | Entwurf: 0 | Status: vom Modulverantwortlichen freigegeben |  
Verantwortlich: Wörzberger

### – Allgemeine Informationen

<b>Anerkannte Lehrveranstaltungen</b>	<u>ARP Wörzberger</u>
---	-----------------------

<b>Gültig ab</b>	Wintersemester 2020/21
------------------	---------------------------

<b>Dauer</b>	1 Semester
--------------	------------

<b>ECTS</b>	5
-------------	---

<b>Zeugnistext (de)</b>	Alternative Rechnerarchitekturen und Programmiersprachen
-------------------------	---

<b>Zeugnistext (en)</b>	Alternate Computer Architectures and Programming Languages
-------------------------	---

<b>Unterrichtssprache</b>	deutsch
---------------------------	---------

<b>abschließende Modulprüfung</b>	Ja
---------------------------------------	----

### Modulprüfung

<b>Benotet</b>	Ja
----------------	----

<b>Konzept</b>	Mündliche Prüfung zu Grundwissen, Anwendung und Anwendbarkeit der vorgestellten Konzepte
----------------	---

<b>Frequenz</b>	Jedes Semester
-----------------	----------------

## – Allgemeine Informationen

### Inhaltliche Voraussetzungen

### Handlungsfelder

Komplexe Rechner-, Kommunikations- und Eingebettete Systeme sowie komplexe Software-Systeme unter interdisziplinären Bedingungen entwerfen, realisieren und bewerten

Wissenschaftlich arbeiten und wissenschaftliche Erkenntnisse anwenden und erweitern

undefined

### Learning Outcomes

ID	Learning Outcome
LO1	Die Studierenden lernen kennen, wenden an und analysieren verschiedene wichtige Konzepte von Rechnerarchitekturen und Programmiersprachen. Dazu wenden sie für jedes ausgewählte Konzept nach einer kurzen Vorstellung es auf ein selbstgewähltes Beispiel an, wozu sie sich weiteres Wissen über das Konzept erwerben müssen, und analysieren die Vor- und Nachteile des Konzepts in einem Bericht. Damit erlangen sie einen größeren Überblick über verfügbare Architekturen und Programmiersprachen für ihre spätere Tätigkeit als IT-Spezialist, Manager oder in der Forschung.

### Kompetenzen

Kompetenz	Ausprägung
Komplexe Systeme und Prozesse analysieren, modellieren, realisieren, testen und bewerten	diese Kompetenz wird vermittelt
Komplexe Aufgaben selbstständig bearbeiten	diese Kompetenz wird vermittelt
Fachwissen erweitern und vertiefen und Lernfähigkeit demonstrieren	diese Kompetenz wird vermittelt
Probleme wissenschaftlich untersuchen und lösen, auch wenn sie unscharf, unvollständig oder widersprüchlich definiert sind	Voraussetzungen für diese Kompetenz (Wissen,...) werden vermittelt

Wissenschaftliche Ergebnisse und technische Zusammenhänge schriftlich und mündlich darstellen und verteidigen	diese Kompetenz wird vermittelt
---	------------------------------------

---

Situations- und sachgerecht argumentieren	Voraussetzungen für diese Kompetenz (Wissen,...) werden vermittelt
---	---

---

Sich selbst organisieren	Voraussetzungen für diese Kompetenz (Wissen,...) werden vermittelt
--------------------------	---

---

Sprachliche und interkulturelle Fähigkeiten anwenden	Voraussetzungen für diese Kompetenz (Wissen,...) werden vermittelt
--	---

---

Projekte organisieren und im Team bearbeiten	diese Kompetenz wird vermittelt
--	------------------------------------

## – Vorlesung / Übungen

<b>Typ</b>	Vorlesung / Übungen
------------	---------------------

<b>Separate Prüfung</b>	Ja
-------------------------	----

<b>Exemplarische inhaltliche Operationalisierung</b>	Die Studierenden lernen weitere, typischerweise aus Zeitgründen nicht im Bachelor erworbenen Konzepte der technischen Informatik kennen. Sie lernen alternative Verarbeitungsmodelle in Hardware ("alternative Rechnermodelle") und Software ("alternative Programmiersprachen") kennen. Sie erkennen auch die Wechselbeziehung zwischen Hardware und Software.
--	---

### Separate Prüfung

<b>Benotet</b>	Nein
----------------	------

<b>Frequenz</b>	Einmal im Jahr
-----------------	----------------

<b>Voraussetzung für Teilnahme an Modulprüfung</b>	Ja
--	----

<b>Konzept</b>	Präsenzübung und Selbstlernaufgaben
----------------	-------------------------------------

## – Praktikum

<b>Typ</b>	Praktikum
------------	-----------

<b>Separate Prüfung</b>	Ja
-------------------------	----

<b>Exemplarische inhaltliche Operationalisierung</b>	Für jedes der vom Dozenten ausgewählten Konzepte gewinnen die Studierenden praktische Erfahrungen in dessen Anwendung auf eine Aufgabenstellung. Dabei erwerben sie sich über die vorgegebenen Materialien hinaus weitere Kenntnisse zum jeweiligen Konzept, entwickeln selbstständig eine Lösung für eine Beispielanwendung, die sie i.d.R. selbst auswählen. Zum Abschluss präsentieren sie ihre Lösungen und analysieren die Lösung sowie die eingesetzten Mittel im Kontext ihrer Informatik-Kenntnisse. Sie organisieren sich in Teams und arbeiten im Rahmen eines vorgegebenen Zeitraums selbstorganisiert an der jeweiligen Aufgabenstellung.
--	---

### Separate Prüfung

<b>Benotet</b>	Ja
----------------	----

<b>Frequenz</b>	Einmal im Jahr
-----------------	----------------

<b>Gewicht</b>	50
----------------	----

<b>Bestehen notwendig</b>	Ja
---------------------------	----

<b>Voraussetzung für Teilnahme an Modulprüfung</b>	Ja
--	----

<b>Konzept</b>	Die Studierenden diskutieren ihre Lösung in einem Kurzvortrag mit anschließender Disputation und beschreiben in einem schriftlichen Projektbericht das Lösungskonzept. Sie nutzen ihre gewonnenen Erfahrungen zur Analyse des Konzepts, insbes. seine Anwendbarkeit auf die Aufgabenstellung.
----------------	---

