

Modul

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: Würzberger

^ Allgemeine Informationen

Anerkannte Lehrveranstaltungen	ARP_Würzberger
Dauer	1 Semester
ECTS	5
Zeugnistext (de)	Alternative Rechnerarchitekturen und Programmiersprachen
Zeugnistext (en)	Alternate Computer Architectures and Programming Languages
Unterrichtssprache	deutsch
abschließende Modulprüfung	Ja

Modulprüfung

Benotet	Ja
Frequenz	Jedes Semester

Prüfungskonzept

Mündliche Prüfung zu Grundwissen, Anwendung und Anwendbarkeit der vorgestellten Konzepte

^ Allgemeine Informationen

Inhaltliche Voraussetzungen

Kompetenzen

Kompetenz	Ausprägung
Komplexe Systeme und Prozesse analysieren, modellieren, realisieren, testen und bewerten	diese Kompetenz wird vermittelt
Komplexe Aufgaben selbständig 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

Exemplarische inhaltliche Operationalisierung

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

Benotet	Nein
Frequenz	Einmal im Jahr

Prüfungskonzept

Präsenzübung und Selbstlernaufgaben

^ Praktikum

Exemplarische inhaltliche Operationalisierung

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

Benotet	Ja
Frequenz	Einmal im Jahr
Gewicht	50
Bestehen notwendig	Ja
Voraussetzung für Teilnahme an Modulprüfung	Ja

Prüfungskonzept

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.