

Modulhandbuch PI1

Praktische Informatik 1

Bachelor Elektrotechnik 2020

Version: 4 | Letzte Änderung: 10.09.2019 15:30 | Entwurf: 0 | Status: vom Modulverantwortlichen freigegeben |
Verantwortlich: Rosenthal

– Allgemeine Informationen

Anerkannte Lehrveranstaltungen	<u>PI1 Rosenthal</u>
Gültig ab	Wintersemester 2020/21
Fachsemester	1
Dauer	1 Semester
ECTS	5
Zeugnistext (de)	Praktische Informatik 1
Zeugnistext (en)	Practical Informatics 1
Unterrichtssprache	deutsch
abschließende Modulprüfung	Ja

Modulprüfung

Benotet	Ja
Konzept	Die Studierenden sollen in einer schriftlichen Klausur folgende Kompetenzen nachweisen, die die o.a. Einzelkompetenzen integrieren: 1.) Sicherer Umgang mit grundlegenden Begrifflichkeiten, 2.) Anwendung programmiersprachlicher und abstrakterer Konstrukte zur Lösung von Anwendungsproblemen, 3.) Prüfung programmiersprachlicher Lösungsvorschläge 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 Struktogramme und Programmstücke und zu 3.) das Finden von Fehlern in vorgegebenen Programmstücken.
Frequenz	Jedes Semester

– Allgemeine Informationen

Inhaltliche Voraussetzungen

Handlungsfelder

Forschung: Von Ansätzen der Grundlagenforschung bis hin zur Industrieforschung. Entwicklung: Algorithmen, Software, Verfahren, Geräte, Komponenten und Anlagen.

Qualitätskontrolle von Produkten und Prozessen, Mess- und Prüftechnologien, Zertifizierungsprozesse.

Koordination kleiner Arbeitsgruppen, international verteilt arbeitender Teams, Koordination von Planungs- und Fertigungsprozessen, sowie Produktmanagement.

Learning Outcomes

ID	Learning Outcome
LO1	<p>Was: Das Modul vermittelt grundlegende Kenntnisse zur Nutzung von Programmiersprachen und entsprechender abstrakter Darstellungsformen bei der algorithmischen Lösung von Anwendungsproblemen. Die Studierenden sollen in die Lage versetzt werden, einschlägige Begriffe und Techniken im praktischen Programmierumfeld sicher anzuwenden: Aufbauend auf den in der Vorlesung vermittelten Kenntnissen (K5,K11,K12) analysieren die Studierenden Problemstellungen (K2,K4), entwerfen Lösungswege dazu (K5), implementieren sie mit Hilfe von Standardwerkzeugen (K5) und prüfen sie (K14).</p> <p>Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. In den Übungen und insbesondere im Praktikum bearbeiten die Studierenden in Kleingruppen Programmieraufgaben und verteidigen ihre Lösungen (K11, K12, K16,K19).</p> <p>Wozu: Kenntnisse und Fertigkeiten in der Anwendung von Programmiersprachen sind essentiell für Ingenieure/-innen, insbesondere in Hinblick auf die Realisierung informationstechnischer Systeme (HF1). Durch ihre praktische Programmierarbeit erwerben die Studierenden zudem Erfahrungen, die wichtig sind für die Erfassung von Anforderungen, die Entwicklung von Konzepten zur technischen Lösung und zu ihrer Bewertung (HF2). Die Durchführung im Team mit dem Dozenten als "Auftraggeber" stärkt die Interaktionsfähigkeit der Studierenden (HF 4).</p>

Kompetenzen

Kompetenz	Ausprägung
Abstrahieren	diese Kompetenz wird vermittelt
Erkennen, Verstehen und analysieren technischer Zusammenhänge	diese Kompetenz wird vermittelt
MINT Modelle nutzen	diese Kompetenz wird vermittelt
MINT-Grundwissen benennen und anwenden	diese Kompetenz wird vermittelt
Informationen beschaffen und auswerten	diese Kompetenz wird vermittelt
Komplexe technische Aufgaben im Team bearbeiten	diese Kompetenz wird vermittelt
Lernkompetenz demonstrieren	Voraussetzungen für diese Kompetenz (Wissen,...) werden vermittelt
Arbeitsergebnisse bewerten	Voraussetzungen für diese Kompetenz (Wissen,...) werden vermittelt

– Vorlesung / Übungen

Typ	Vorlesung / Übungen
------------	---------------------

Separate Prüfung	Nein
-------------------------	------

Exemplarische inhaltliche Operationalisierung	Die Lehrveranstaltung führt in grundlegende Konzepte der Programmierung ein und illustriert sie am Beispiel einer höheren Programmiersprache. Die Studierenden sollen lernen, diese Konzepte selbstständig zur Lösung von Anwendungsproblemen einzusetzen.
--	--

– Praktikum

Typ	Praktikum
------------	-----------

Separate Prüfung	Ja
-------------------------	----

Exemplarische inhaltliche Operationalisierung	Die Studierenden schließen sich zu Kleingruppen zusammen. Jede Kleingruppe absolviert mehrere "Praktikumsrunden" mit zugewiesenen Laborterminen. In jeder Runde werden Programmieraufgaben algorithmischer Art gelöst - erstens durch eine abstraktere Darstellung (z.B. Beschreibung eines Algorithmus durch ein Struktogramm), zweitens durch eine programmtechnische Umsetzung (z.B. C-Programm). 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.
--	--

Separate Prüfung

Benötet	Nein
----------------	------

Frequenz	Einmal im Jahr
-----------------	----------------

Voraussetzung für Teilnahme an Modulprüfung	Ja
--	----

Konzept	Wie unter "exemplarische inhaltliche Operationalisierung" dargestellt, findet die Leistungsüberprüfung stufenweise statt (mehrere Runden, in jeder Runde schriftlicher Kurztest, persönliches Gespräch mit dem Betreuer, Programmieren unter Aufsicht). Durch diese stufenweise Vorgehensweise können sämtliche aufgeführte Kompetenzen individuell abgeprüft werden.
----------------	---

