

Lehrveranstaltungshandbuch IP

Informatik Projekt

Version: 2 | Letzte Änderung: 30.09.2019 13:05 | Entwurf: 0 | Status: vom verantwortlichen Dozent freigegeben

– Allgemeine Informationen

Langname	Informatik Projekt
Anerkennende LModule	<u>IP_BaET</u>
Verantwortlich	Prof. Dr. Stefan Kreiser Professor Fakultät IME
Gültig ab	Wintersemester 2020/21
Niveau	Bachelor
Semester im Jahr	Wintersemester
Dauer	Semester
Stunden im Selbststudium	72
ECTS	3
Dozenten	Prof. Dr. Stefan Kreiser Professor Fakultät IME Al Ghouz
Voraussetzungen	Programmierkenntnisse in C
Unterrichtssprache	deutsch, englisch bei Bedarf
separate Abschlussprüfung	Nein

Literatur

Reg. Rechenzentrum der Uni Hannover: Die Programmiersprache C (Campus-IT FH Köln)

S. Kochan: Programming in C (Pearson)

P. Prinz, T. Crawford: C in a Nutshell (O'Reilly)

R.Lischner: C++ in a Nutshell (O'Reilly)

T. DeMarco: Structured Analysis and System Specification (Prentice Hall PTR)

M. Dausmann et. al.: C als erste Programmiersprache (Vieweg, Teubner)

J. Wolf: C von A bis Z, Das umfassende Handbuch (Openbook, Rheinwerk Computing)

– Projekt

Lernziele

Zieltyp	Beschreibung
Fertigkeiten	<p>Studierende zeigen, dass sie in der prozeduralen Programmiersprache C lauffähige Programme entwickeln können, die komplexere, algorithmisch lösbare Aufgabenstellungen nachvollziehbar und vollständig lösen. Dabei können die Studierenden:</p> <ol style="list-style-type: none">1. Algorithmen und Datenstrukturen aus einer textuellen Aufgabenstellung extrahieren und einen C-Programmcode zur Lösung der Aufgabenstellung mit Hilfe einer funktionalen Gliederung und unter Verwendung der extrahierten und ggfs. vorgegebener Algorithmen und Datenstrukturen sowie unter Einhaltung vorgegebener Programmierrichtlinien entwickeln und systematisch prüfen.2. Die Funktion der Software und den Aufbau des Programmcodes erläutern, dokumentieren, begründen und modifizieren.3. Eine integrierte Entwicklungsumgebung sicher zur Erstellung lauffähiger Programme nutzen. <p>Die zu bearbeitenden Projektaufgaben sind je nach Komplexität bzw. Schwierigkeitsgrad einem von drei unterschiedlichen Pools zugeordnet (geringe, mittlere und höhere Komplexität / Bearbeitungsumfang). Je Pool müssen Studierende eine oder mehrere Projektaufgaben bis zu einem vorgegebenen Termin vollständig bearbeiten. Zur Bearbeitung einer Projektaufgabe dürfen / sollen Studierende Projektgruppen je maximal drei Personen bilden. Je Pool müssen Studierende alle zugewiesenen Programme lauffähig abgeben und dann eines ihrer Programme in einem Fachgespräch erläutern, begründen und ggfs. modifizieren.</p>

Besondere Literatur

Tutorials zur Installation der Virtuellen Maschine und zur Nutzung der Entwicklungsumgebung

Besondere Voraussetzungen

eigener Notebook-Computer

Begleitmaterial

- digitale Aufgabensammlung (Ilias)
- Virtuelle Maschine unter Linux zur Programmentwicklung mit Modellierungswerkzeug (Struktogrammeditor) und integrierter Entwicklungsumgebung (Eclipse/C)

Separate Prüfung

Ja

Separate Prüfung

Prüfungstyp

undefined

Aufwand Präsenzlehre

Typ	Präsenzzeit (h/Wo.)
Projekt	1
Tutorium (freiwillig)	0

Details

Studierende zeigen je Pool, dass sie die Lösungen / Programme zu den gegebenen Projektaufgaben unter Einhaltung vorgegebener Programmierrichtlinien in einem kleinen Projektteam erarbeiten und termingerecht fertigstellen können. Sie weisen anhand dokumentierter Ergebnisse aus Programmläufen nach (Tests mit vorgegebenen und sinnvoll ergänzten eigenen Prüfkriterien), dass ihre Programme die jeweils geforderte Funktion erfüllen. Im Fachgespräch weisen die Studierenden nach, dass sie die Programme selbst entwickelt haben. Dazu erläutern und begründen sie die Funktionsfähigkeit des Programmcodes an wesentlichen Abschnitten (z. B. Bedingungsausdrücke in Alternativen oder Iterationen, Funktionsaufrufe, Datentypen und Strukturen) anhand von Testläufen, ggfs. auch mit neuen Prüfkriterien, und sind fähig, an ihrem Programmcode jeweils mindestens eine sinnvolle Modifikation und/oder Erweiterung vorzunehmen und die Ablauffähigkeit des geänderten Programms zu zeigen bzw. nachzuweisen.

Mindeststandard

- Alle Lösungsbestandteile sind vorhanden, d. h. die grundsätzlich zu jeder Aufgabe geforderten und die in der bearbeiteten Aufgabe speziell geforderten Lösungsbestandteile.
- Die Programmierrichtlinien wurden eingehalten.
- Studierende(r) kann den implementierten Algorithmus in seinen wesentlichen Teilen nachvollziehbar erläutern und kann für wesentliche Funktionen, Kontroll- und Datenstrukturen deren Verwendung im Algorithmus anhand der Funktionsaufrufe, Bedingungsausdrücke und Speicherverwaltung erläutern und nachvollziehbar begründen.
- Studierende(r) kann die in der Aufgabenstellung geforderte Funktion des Programms an mindestens einem wesentlichen Abschnitt des Programmcodes diskutieren, begründen und in Testläufen nachweisen.
- Studierende(r) kann das Programm gemäß Anforderung im Fachgespräch in einem algorithmisch relevanten Umfang überarbeiten, d.h. signifikante Änderungen im Programmcode vornehmen und die Funktionsfähigkeit des geänderten Programms begründen.