

Modulbezeichnung	Programmierung I
Modulbezeichnung (eng.)	Programming I
Semester (Häufigkeit)	1 (jedes Wintersemester)
ECTS-Punkte (Dauer)	5 (1 Semester)
Art	Pflichtmodul
Studentische Arbeitsbelastung	15 h Kontaktzeit + 135 h Selbststudium
Voraussetzungen (laut BPO)	Keine
Empf. Voraussetzungen	Keine
Verwendbarkeit	BORE
Prüfungsart und -dauer	Einsendeaufgaben (Praktikum) als Studienleistung Test am Rechner (2h) oder mündliche Prüfung (30 Minuten)
Lehr- und Lernmethoden	Multimedial aufbereitetes Online-Studienmodul zum Selbststudium mit zeitlich parallel laufender Online-Betreuung und regelmäßigen virtuellen Lehrveranstaltungen
Modulverantwortliche(r)	J. Mäkiö
<p>Voraussetzungen für die Vergabe von Leistungspunkten</p> <p>Studienleistung (Praktikum): 3 Einsendeaufgaben (10h - Dauer) während des Semesters oder danach einzureichen (1 CP). Bewertet mit 'Bestanden', wenn 60% i.O. sind. Abgabe per Quellcode - Upload. Prüfungsleistung (4 CP): Bearbeitung von Programmier-Aufgaben als 'Test am Rechner' (im Prüfungszeitraum, online von zu Hause aus) Dauer: 2 Stunden. Jeweils Abgabe eines kommentierten Quellcodes als Upload. Identitätskontrolle und Beobachtung per Kamera während der Arbeit. Hilfsmittel: Kopfwissen, Papier und Stift, Entwicklungsumgebung mit PC. Oder mündliche Prüfung (30 Minuten).</p>	
<p>Qualifikationsziele</p> <p>Die Studierenden</p> <ul style="list-style-type: none"> • können die Funktionsweise und die Zusammenarbeit der zentralen Software- und Hardware-Komponenten eines Rechners erklären. • können einen vorgegebenen C-Programmcode analysieren und das Ergebnis dessen bestimmen. • können einfache Programmieraufgaben in C selbstständig lösen. • können den Aufbau eines C-Programms wiedergeben und erläutern. • können die verschiedenen Zeigertypen in C erläutern und diese in einfachen Aufgaben richtig einsetzen. • können die Programmsteuerungsstrukturen in C in einfachen Programmieraufgaben richtig anwenden sowie die wesentlichen Unterschiede dieser einem dritten gegenüber erläutern. • können Funktionen in C vereinbaren, aufrufen und einen Funktionswert übergeben. • können strukturierte Datentypen in C definieren und korrekt in einfachen Aufgaben einsetzen. 	

Lehrinhalte

Grundbegriffe der Informatik Semantik und Syntax; Algorithmus; Analog und Digital

Hardwarekomponenten Das EVA-Prinzip; Rechneraufbau

Programmieren Mit Editor und Compiler; Mit einer Entwicklungsumgebung

Elementare Datentypen, Variablen und Zuweisungen Programmstruktur; Variablenkonzept; Elementare Datentypen

Elementare Operatoren und Programmiersteuerungsstrukturen Operatoren und Operanden; Auswertungsreihenfolge; Elementare Operatoren; Implizite Typumwandlung; Programmiersteuerungsstrukturen

Daten Ein- und Ausgabe Konzept; Standardeingabe und -ausgabe; Bibliotheken zur Ein- und Ausgabe; High-Level-Funktionen für die Standardein- und -ausgabe; Funktionen für die Ein- und Ausgabe in eine Datei

Operieren mit Zeigern und Arrays Zeigervariablen; Eindimensionale Arrays; Zeichenketten

Strukturierte Datentypen Strukturen; Unionen; Aufzählungen; Vereinbarung eigener Typnamen

Fortgeschrittenes Operieren mit Zeigern Arrays; Zeigerarithmetik; Dynamische Speicherverwaltung

Funktionen Definition von Funktionen; Aufruf von Funktionen; Rücksprung mit der return-Anweisung; Zeiger als Funktionsparameter; Das Hauptprogramm als Funktion; Deklaration von Funktionen und Header-Dateien; Funktions-Bibliotheken

Literatur

Gaicher, Heimo (2012): Programmieren in C. Hamburg: tredition.

Theis, Thomas (2014): Einstieg in C. 1. Aufl. Bonn: Galileo Press (Galileo computing).

Wolf, Jürgen (2015): C von A bis Z. 3., aktualis. u. erw. Aufl. 2009, 4. korr. Nachdr. 2015. Bonn: Rheinwerk (Rheinwerk Computing).

Lehrveranstaltungen

Dozenten/-innen	Titel der Lehrveranstaltung
M. Masur	Programmierung I
M. Masur	Praktikum Programmierung I