

6 Programmierung I	
Semester	1
Dauer (Semester)	einsemestrig
Credit Points	5
Pflicht/ Wahlpflicht	Pflicht
Häufigkeit des Angebotes/ Verwendbarkeit	Jedes Wintersemester
Modulverantwortliche(r)	Prof. Dr. Juho Mäkiö, Hochschule Emden/Leer
Modulverantwortliche(r) standortbezogen	Markus Masur
Lerngebiet	Grundlagen der Informationstechnik
Teilnahmevoraussetzungen	keine
Lernergebnisse	<p>Die Studierenden</p> <ul style="list-style-type: none"> <li>• können die Funktionsweise und die Zusammenarbeit der zentralen Software- und Hardware-Komponenten eines Rechners erklären.</li> <li>• können einen vorgegebenen C-Programmcode analysieren und das Ergebnis dessen bestimmen.</li> <li>• können einfache Programmieraufgaben in C selbstständig lösen.</li> <li>• können den Aufbau eines C-Programms wiedergeben und erläutern.</li> <li>• können die verschiedenen Zeigertypen in C erläutern und diese in einfachen Aufgaben richtig einsetzen.</li> <li>• können die Programmsteuerungsstrukturen in C in einfachen Programmieraufgaben richtig anwenden sowie die wesentlichen Unterschiede dieser einem dritten gegenüber erläutern.</li> <li>• können Funktionen in C vereinbaren, aufrufen und einen Funktionswert übergeben.</li> <li>• Die Studierenden können strukturierte Datentypen in C definieren und korrekt in einfachen Aufgaben einsetzen.</li> </ul>
Prüfungsvorleistung	keine
Medien-/ Lernform	Multimedial aufbereitetes Online-Studienmodul zum Selbststudium mit zeitlich parallel laufender Online-Betreuung (E-Mail, Foren, Chat, Webkonferenzen, Einsendeaufgaben u. a.) sowie dazugehörigem Programmier-Labor in Online-Form
Arbeitsaufwand	<p>Selbststudium: ca. 132 h</p> <p>Webkonferenzteilnahme: ca. 16 h</p> <p>Prüfung: 120 Minuten</p>
Präsenzart	In Online-Konferenz möglich
Präsenzinhalte	Programmierübungen

Prüfungsform	Klausur (120 min.)
Voraussetzung für die Vergabe von Leistungspunkten	Studienleistung (Programmierlabor): 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).
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).
weitere Hinweise	Dieses Modul wird auf Deutsch angeboten

Studieninhalte
<p><b>Grundbegriffe der Informatik</b> Semantik und Syntax; Algorithmus; Analog und Digital</p> <p><b>Hardwarekomponenten</b> Das EVA-Prinzip; Rechneraufbau</p> <p><b>Programmieren</b> Mit Editor und Compiler; Mit einer Entwicklungsumgebung</p> <p><b>Elementare Datentypen, Variablen und Zuweisungen</b> Programmstruktur; Variablenkonzept; Elementare Datentypen</p> <p><b>Elementare Operatoren und Programmiersteuerungsstrukturen</b> Operatoren und Operanden; Auswertungsreihenfolge; Elementare Operatoren; Implizite Typumwandlung; Programmiersteuerungsstrukturen</p> <p><b>Daten Ein- und Ausgabe</b> 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</p> <p><b>Operieren mit Zeigern und Arrays</b> Zeigervariablen; Eindimensionale Arrays; Zeichenketten</p> <p><b>Strukturierte Datentypen</b> Strukturen; Unionen; Aufzählungen; Vereinbarung eigener Typnamen</p> <p><b>Fortgeschrittenes Operieren mit Zeigern</b> Arrays; Zeigerarithmetik; Dynamische Speicherverwaltung</p> <p><b>Funktionen</b> Definition von Funktionen; Aufruf von Funktionen; Rücksprung mit der return-Anweisung; Zeiger als</p>

Funktionsparameter; Das Hauptprogramm als Funktion; Deklaration von Funktionen und Header-Dateien; Funktions-Bibliotheken