

<p>Modultitel:</p> <p>Modultyp:</p> <p>Englische Übersetzung:</p>	<p>Technische Informatik 1 – Modulteil Einführung in das Programmieren in C</p> <p>Pflichtmodul im 1. Fachsemester</p> <p>Computer science 1 – submodule „introduction into programming in C“</p>
<p>Qualifikationsziele</p> <p>Die Gliederung der Kompetenzbereiche erfolgt analog der Gliederung des Qualifikationsrahmens für deutsche Hochschulabschlüsse (HQR, 2017)</p>	<p>Gesamtqualifikationsziel</p> <p>Die Studierenden erwerben ein Grundlagenwissen über Programmieren in C als prozessnahe Programmiersprache und als Grundlage für das Erlernen objektorientierter Programmiersprachen wie C. Es wird dabei das algorithmische Denken geschult: Regelmäßigkeiten in Abläufen entdecken und formulieren können, Abstraktionen formulieren können. Im Rahmen von praktischen Übungen wird die Anwendung der Programmierfähigkeiten vertieft.</p> <p>Fachkompetenz (Wissen und Verstehen)</p> <ul style="list-style-type: none"> • Struktur und Aufbau von C-Programmen • Abstraktion von Problemstellungen mit Hilfe von Nassi-Shneiderman-Programmen • Grundlegende Syntax der Programmiersprache C • Programmierung mit imperativen Programmiersprachen nach dem EVA-Prinzip • Grundlagen der Zeigerprogrammierung <p>Methodenkompetenz (Einsatz, Anwendung und Erzeugung von Wissen)</p> <ul style="list-style-type: none"> • Umsetzung einfacher Algorithmen zur Automatisierung von Abläufen. <p>Sozialkompetenz (Kommunikation und Kooperation)</p> <ul style="list-style-type: none"> • Mitarbeit in einem Programmierteam <p>Selbstkompetenz (Wissenschaftliches Selbstverständnis /Professionalität)</p> <p>Selbstständige Entwicklung und Programmierung einfacher Algorithmen</p>
<p>Inhalte</p>	<ul style="list-style-type: none"> • Ursprünge und Konzepte von Programmiersprachen • Modellieren von Programmen z.B. nach Nassi-Shneiderman • Compiler, Compilervorgang • Grundaufbau eines C-Programms, Header-Dateien • Ein- und Ausgabe-Anweisungen • Datentypen • Relationale Operatoren • Bedingungsanweisungen if, if...else, switch • Schleifenanweisungen do, do...while, for (vollständig und verkürzt) • Ein- und mehrdimensionale Datenfelder (arrays), Zeiger • Programmierung von logischen und mathematischen Funktionen • Unterprogramme / Funktionen • Algorithmen wie <ul style="list-style-type: none"> ○ Bubblesort-Sortieralgorithmus

Modulhandbuch des B.Sc. Wirtschaftsingenieurwesen

	<ul style="list-style-type: none"> ○ Einfache grafische Ausgaben im Konsolen-Fenster ○ Numerisches Lösen von Differenzialgleichungen, numerisches Integrieren, numerisches Differenzieren.
Lehrformen	Vorlesungen (2 SWS) mit seminaristischen Anteilen, Übungen und Laboranteilen
Unterrichtssprache	Deutsch, bei parallelen Teilungsgruppen ist bei Bedarf eine Teilungsgruppe in englischer Sprache möglich
Voraussetzungen für die Teilnahme	<p>Empfohlen: PC-Bedienung, Abiturwissen in Mathematik</p> <p>Erforderlich: -</p>
Verwendbarkeit des Moduls	Dieses Modul ist Pflichtmodul und Teil eines Wahlpflichtcurriculums Technische Informatik: TI 1 – Grundlagen der Technischen Informatik, Grundlagen des Programmierens in C, TI 2 – Softwareengineering, TI 3 (Embedded Systems/Microcontroller).
Art, Voraussetzung und Sprache der Modulprüfung	<p>Regelmäßige Prüfungsform für die Modulprüfung: PL Abschlussprüfung in Form einer Klausur von 1 Stunde Dauer, Klausurvoraussetzung: Testate, die mit 50% bestanden werden müssen.</p> <p>Weitere mögliche Prüfungsformen: -</p> <p>Bei mehr als einer möglichen Prüfungsform im Modul wird die zu erbringende Prüfungsform von dem verantwortlichen Lehrenden zu Beginn der Lehrveranstaltung bekannt gegeben.</p>
Gesamtarbeitsaufwand	<p>3 Leistungspunkte (LP)</p> <p>2 Semesterwochenstunden (SWS)</p> <p>Gesamtarbeitsaufwand 90 h, davon Präsenzstudium 28 h und Selbststudium 62 h</p>
Häufigkeit des Angebots	Jedes Semester
Dauer	1 Semester
Literatur	Goll/Bröckl/Dausmann: C als erste Programmiersprache. SpringerVieweg