

## 2.2 Modul Informatik II

Modulbezeichnung	<b>Informatik II</b>
Kürzel für Stundenplan	Inf2
Semester	2
Modulverantwortliche(r)	Prof. Dr.-Ing. Uwe Krohn
Dozent(in)	Prof. Dr.-Ing. Uwe Krohn
Sprache	Deutsch
Zuordnung zum Curriculum	INF – Pflichtmodul
Lehrform / SWS	4 V, 2 Pr
Arbeitsaufwand	64 h Vorlesung Präsenz + 32 h Praktikum Präsenz + Vor- und Nachbearbeitung =210 h
Kreditpunkte (gem. ECTS)	7
Voraussetzungen	Informatik I/ Programmieren I
Lernziele / Kompetenzen	<p>Die Studierenden kennen effiziente Algorithmen und Datenstrukturen aus den wichtigsten Gebieten der Angewandten Informatik.</p> <p>Sie beherrschen die notwendigen theoretischen Grundlagen: Abstrakte Datentypen, Berechenbarkeit sowie Algorithmenparadigmen.</p> <p>Die Studierenden sind in der Lage, Algorithmen und Datenstrukturen zu verstehen, zu entwerfen und zu implementieren. Sie können Algorithmen und Datenstrukturen bzgl. ihrer Laufzeit und ihres Speicherverbrauchs analysieren und auf Basis dieser Analysen vergleichen.</p> <p>Die Studierenden verfügen über Kompetenzen zur Auswahl und Anwendung geeigneter Datenstrukturen und Algorithmen in der Softwareentwicklung zur Verbesserung der Effizienz und der Wartbarkeit der Programme sowie zur Verkürzung der Entwicklungszeit.</p>
Gender-/Diversity	<p>Die folgenden Gender-/Diversity-bezogenen Aspekte werden berücksichtigt:</p> <ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> Verwendung geschlechtergerechter Sprache</li> <li><input checked="" type="checkbox"/> Einbezug unterschiedlicher didaktischer Methoden</li> <li><input type="checkbox"/> Sichtbarmachen von Vielfalt im Fach (z.B. Nennung von Forscherinnen oder Beispielen aus anderen Kulturen)</li> <li><input type="checkbox"/> Inhaltlicher Einbezug von Gender-/Diversity-Aspekten in der Lehre</li> </ul>

Modulbezeichnung	Informatik II
Inhalt	<p><b>Abstrakte Datentypen (ADT)</b></p> <ul style="list-style-type: none"> <li>• Spezifikation von Datenstrukturen</li> <li>• Datenstrukturen als Algebren</li> <li>• Information Hiding, Modularisierung</li> <li>• Spezifikation von ADTs</li> </ul> <p><b>Stacks</b></p> <ul style="list-style-type: none"> <li>• ADT-Spezifikation</li> <li>• Implementierung mit Array</li> <li>• Implementierung mit verketteter Liste</li> <li>• Auswertung von Postfix-Ausdrücken mit einem Stack</li> </ul> <p><b>Queues</b></p> <ul style="list-style-type: none"> <li>• ADT-Spezifikation</li> <li>• Implementierung mit Array</li> <li>• Implementierung mit verketteter Liste</li> </ul> <p><b>Listen</b></p> <ul style="list-style-type: none"> <li>• Implementierung mit verketteter Liste</li> <li>• Iteratoren</li> <li>• Doppelt verkettete Listen</li> </ul> <p><b>Bäume</b></p> <ul style="list-style-type: none"> <li>• Binäre Bäume</li> <li>• Suchbäume</li> <li>• Baum-Traversierungen</li> <li>• Visitor-Pattern</li> <li>• Balancierte Bäume</li> <li>• AVL-Bäume</li> <li>• 2-3-4-Bäume</li> <li>• B-Bäume, B<sup>+</sup>-Bäume</li> <li>• Heap, HeapSort, Prioritätswarteschlangen</li> </ul> <p><b>Graphen</b></p> <ul style="list-style-type: none"> <li>• Ungerichtete Graphen</li> <li>• Gerichtete Graphen</li> <li>• Gewichtete Graphen</li> <li>• Realisierung von Graphen</li> <li>• Breitendurchlauf</li> <li>• Tiefendurchlauf</li> <li>• Minimale Spannbäume</li> <li>• Kürzeste Wege</li> </ul> <p><b>Automatentheorie</b></p> <ul style="list-style-type: none"> <li>• Endliche Automaten</li> <li>• Petrinetze</li> </ul>

Modulbezeichnung	<b>Informatik II</b>
Literatur	<ul style="list-style-type: none"> <li>• H. P. Gumm, M. Sommer: Einführung in die Informatik, Oldenbourg</li> <li>• Th. H. Cormen, Ch. E. Leiserson, R. Rivest, C. Stein: Algorithmen – Eine Einführung</li> </ul>
Studien-/Prüfungsleistungen	Projektarbeit