

| 13 Algorithmen und Datenstrukturen Algorithms and Data Structures | |
|--|---|
| Semester | 3 |
| Dauer (Semester) | einsemestrig |
| Credit Points | 5 |
| Pflicht/ Wahlpflicht | Pflicht |
| Häufigkeit des Angebotes/ Verwendbarkeit | im Aufnahmerhythmus |
| Modulverantwortliche(r) | Dr. Jörg Weimar |
| Lerngebiet | Informatik: Algorithmen und Datenstrukturen |
| Teilnahmevoraussetzungen | Mathematische Grundlagen, Programmierung |
| Lernergebnisse | <p>Die Studierenden...</p> <ul style="list-style-type: none"> • lernen Algorithmen und Datenstrukturen und die darauf angewendeten Techniken zur Verifikation und zur Analyse ihrer Komplexität kennen. • verstehen Such- und Sortieralgorithmen und Speicher- und Zugriffstechniken von bzw. auf Listen, Bäume und Hashtabellen. • verstehen Methoden zur Komplexitätsanalyse von Algorithmen. • können Algorithmen und Datenstrukturen in konkreten Anwendungssystemen zur Lösung einer gestellten Anforderung anwenden und beherrschen. • können Algorithmen verifizieren und bezüglich ihrer Zeit- und Platzkomplexität analysieren. • können Algorithmen und Datenstrukturen weiterentwickeln, um konkrete Probleme zu lösen. • können Algorithmen und Datenstrukturen bezüglich ihrer Zeit- und Platzkomplexität und weiterer Leistungskriterien bewerten und für ihre konkrete Anwendung auswählen. |
| Prüfungsvorleistung | Pflicht-Präsenzteilnahme (4 x 45 Minuten) |
| Medien-/ Lernform | Multimedial aufbereitetes Online-Studienmodul zum Selbststudium mit zeitlich parallel laufender Online-Betreuung (E-Mail, Foren, Chat, Webkonferenzen, Einsendeaufgaben u. a.) sowie ggf. Präsenzphasen |
| Arbeitsaufwand | Präsenzteilnahme: ca. 3 h Prüfung: 120 Minuten Selbststudium: 119 h Betreutes Lernen: 31 h Vorbereitung PVL: 0 h |
| Präsenzart | erfordert physische Anwesenheit |

| | |
|------------------|--|
| Präsenzinhalte | Besprechung inhaltlicher Fragen zum Studienmodul Besprechung ausgewählter Übungsaufgaben und gemeinsame Bearbeitung weiterer Beispiele Klärung sonstiger Fragen Klausurvorbereitung |
| Prüfungsform | Klausur (120 min.) |
| Literatur | Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L.: Algorithmen - eine Einführung, 4. Auflage. Olden-bourg Verlag, München 2013. ISBN 978-3-486-74861-1 Baase, Sara; van Geldern, Allen: Computer Algorithms - Introduction to Design and Analysis, 3rd Edition. Addison Wesley Longman Inc., Mass. 2000. ISBN 0-201-612244-5 Schöning, Uwe: Algorithmik. Spektrum Akademischer Verlag, Heidelberg. 2001. ISBN 3-8274-1092-4 |
| weitere Hinweise | Dieses Modul wird auf Deutsch angeboten |

| Studieninhalte | |
|--|--|
| <p>Das Studienmodul gibt eine Einführung in das Fach Algorithmen und Datenstrukturen. Das Ziel dabei ist einerseits, einige Algorithmen und einige Datenstrukturen kennenzulernen und sie zu verstehen. Im Vordergrund stehen Such- und Sortieralgorithmen und die dynamische Datenstrukturen Listen, Bäume und Hashtabellen. Alle Algorithmen werden in so genanntem Pseudocode dargestellt. Darüber hinaus geht es aber auch um die Analyse von Algorithmen. Eine Technik zu deren Verifikation wird kurz eingeführt, die Verfahren zur Bestimmung ihrer Komplexität bzgl. Laufzeit und Speicherplatz werden dagegen tiefergehend diskutiert. Hierfür werden einige Komplexitätsmaße eingeführt und diese auf alle vorgestellten Algorithmen angewendet.</p> <p>Die Studierenden sollen die Algorithmen und Datenstrukturen und die darauf angewandten Analysetechniken kennen lernen und verstehen, sie in ihren fachlichen Kontext einordnen und in konkreten Problemen anwenden können.</p> | |
| <ol style="list-style-type: none"> 1. Einleitung <ol style="list-style-type: none"> 1.1 Was ist ein Algorithmus 1.2 Darstellung von Algorithmen 2. Analyse von Algorithmen <ol style="list-style-type: none"> 2.1 Verifikation 2.2 Komplexität 2.3 Asymptotische Notation 2.4 Optimalität 3. Rekursion Arbeitsaufwand ca. 10 h <ol style="list-style-type: none"> 3.1 Lineare Rekursion 3.2 Divide and Conquer | |

4. Suchen und Sortieren
 - 4.1 Problemspezifikation
 - 4.2 Sequentielles Suchen
 - 4.3 Binäres Suchen
 - 4.4 Suchen und Optimalität
 - 4.5 Bubble-Sort
 - 4.6 Merge-Sort
 - 4.7 Quick-Sort
 - 4.8 Sortieren und Optimalität
 - 4.9 Sortieren durch Abzählen

5. Dynamische Datenstrukturen
 - 5.1 Abstrakte Datentypen
 - 5.2 Verkettete Listen
 - 5.3 Binäre Bäume
 - 5.4 Binäre Heaps
 - 5.4.1 Konstruktion und Erhalten eines Heaps
 - 5.4.2 Heap-Sort
 - 5.4.3 Prioritäts-Warteschlangen

6. Hashverfahren Datenstrukturen
 - 6.1 Adresstabelle mit direktem Zugriff
 - 6.2 Hashtabellen
 - 6.3 Hashfunktionen
 - 6.4 Offene Adressierung
 - 6.5 Array Doubling