

## Course: Strategien kompetitiver Programmierung (IDL)

General information	
<b>Course Name</b>	Strategien kompetitiver Programmierung (IDL) Strategies for competitive programming (IDL)
<b>Course code</b>	COCO-IDL
<b>Lecturer(s)</b>	Oenings, Hendrik (hendrik.oenings@haw-kiel.de)
<b>Occurrence frequency</b>	Regular
<b>Module occurrence</b>	In der Regel jedes Semester
<b>Language</b>	Deutsch

Qualification outcome
<i>Areas of Competence: Knowledge and Understanding; Use, application and generation of knowledge; Communication and cooperation; Scientific self-understanding / professionalism.</i>
Die Studierenden - kennen grundlegende Algorithmen und Datenstrukturen. - verstehen die Bedeutung von Laufzeit-/Speicherkomplexität.
Die Studierenden - können gegebene Problemstellungen auf bekannte algorithmische Methoden reduzieren und diese anpassen. - können den Umfang eines Problems einschätzen und beurteilen, welche algorithmischen Strategien effizient einsetzbar sind. - können ungefähr einschätzen, wie aufwändig die Lösung eines Problems in Bezug auf Laufzeit und Implementierungsaufwand ist.
Die Studierenden - können im Team mit anderen ihre Ideen kommunizieren und erarbeiten. - können aus natürlichsprachlichen Texten und Beispielen abstrahieren und Problemstellungen erkennen.
Die Studierenden - können selbstständig komplexe Aufgabenstellungen einschätzen und Lösungsstrategien entwickeln. - können ihre Ergebnisse reflektieren und bewerten und Grenzen ihrer eingesetzten Methode in Bezug auf Speicherbedarf und Laufzeit in der Praxis einschätzen. - können bei komplexen Aufgabenstellungen beurteilen, welche Lösungsstrategie(n) unter Berücksichtigung sowohl der Problemgröße und Laufzeit als auch des Implementierungsaufwands einzusetzen sind.

Content information	
Content	<ul style="list-style-type: none"><li>- Bestimmung der Laufzeitkomplexität eines Algorithmus</li><li>- Datenstrukturen (dynamisches Array, Set, Map, ...)</li><li>- Backtracking</li><li>- Greedy-Algorithmen</li><li>- Dynamische Programmierung</li><li>- Range-Queries</li><li>- Graph-Algorithmen (DFS/BFS, Shortest path, Spanning tree, Ford-Fulkerson, ...)</li><li>- String-Algorithmen</li></ul>
Literature	A. Laaksonen: Competitive Programmer's Handbook ( <a href="https://cses.fi/book/book.pdf">https://cses.fi/book/book.pdf</a> )

Teaching format of this course	
Teaching format	SWS
Lehrvortrag + Übung	2

<b>Examinations</b>	
<b>COCO-IDL - Technischer Test</b>	Method of Examination: Technischer Test Duration: 30 Minutes Weighting: 0% wird angerechnet gem. § 11 Satz 2 PVO: No Graded: No Remark: Die Studierenden beantworten Wissensfragen zu den in der Vorlesung behandelten Themen und wenden ihr Wissen auf ein Beispielproblem an.
<b>Ungraded Course Assessment</b>	Yes

<b>Miscellaneous</b>	
<b>Miscellaneous</b>	Für die Teilnahme sind Programmierkenntnisse hilfreich, aber nicht erforderlich.  Studierende, welche diese Lehrveranstaltung als IDL-Veranstaltung einbringen, können nicht gleichzeitig das Modul COCO einbringen.