

## 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 im Sommersemester
<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	<ul style="list-style-type: none"> <li>- kennen grundlegende Algorithmen und Datenstrukturen.</li> <li>- verstehen die Bedeutung von Laufzeit-/Speicherkomplexität.</li> </ul>
Die Studierenden	<ul style="list-style-type: none"> <li>- können gegebene Problemstellungen auf bekannte algorithmische Methoden reduzieren und diese anpassen.</li> <li>- können den Umfang eines Problems einschätzen und beurteilen, welche algorithmischen Strategien effizient einsetzbar sind.</li> <li>- können ungefähr einschätzen, wie aufwändig die Lösung eines Problems in Bezug auf Laufzeit und Implementierungsaufwand ist.</li> </ul>
Die Studierenden	<ul style="list-style-type: none"> <li>- können im Team mit anderen ihre Ideen kommunizieren und erarbeiten.</li> <li>- können aus natürlichsprachlichen Texten und Beispielen abstrahieren und Problemstellungen erkennen.</li> </ul>
Die Studierenden	<ul style="list-style-type: none"> <li>- können selbstständig komplexe Aufgabenstellungen einschätzen und Lösungsstrategien entwickeln.</li> <li>- können ihre Ergebnisse reflektieren und bewerten und Grenzen ihrer eingesetzten Methode in Bezug auf Speicherbedarf und Laufzeit in der Praxis einschätzen.</li> <li>- können bei komplexen Aufgabenstellungen beurteilen, welche Lösungsstrategie(n) unter Berücksichtigung sowohl der Problemgröße und Laufzeit als auch des Implementierungsaufwands einzusetzen sind.</li> </ul>

Content information	
<b>Content</b>	<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>
<b>Literature</b>	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	
<b>Teaching format</b>	<b>SWS</b>
Lehrvortrag + Übung	2

<b>Examinations</b>	
<b>Ungraded Course Assessment</b>	Yes

<b>Miscellaneous</b>	
<b>Miscellaneous</b>	<p>Für die Teilnahme sind Programmierkenntnisse hilfreich, aber nicht erforderlich.</p> <p>Studierende, welche diese Lehrveranstaltung als IDL-Veranstaltung einbringen, können nicht gleichzeitig das Modul COCO einbringen.</p> <p>Bei erfolgreicher Teilnahme an dieser Lehrveranstaltung erfolgt eine Anrechnung bei den damit verknüpften Modulen (z. B. WIL1, WIL2) mit 2,5 Leistungspunkten.</p>