HAW Kiel
Hochschule für Angewandte Wissenschaften Kiel
Kiel University of Applied Sciences

BI140 - Neben- und Parallelläufigkeit mit C++
BI140 - Concurrency and Parallelism using C++

General information	
Module Code	BI140
Unique Identifier	NebenPLCPP-01-BA-M
Module Leader(s)	Prof. Dr. Manzke, Robert (robert.manzke@haw-kiel.de) Greve, Thomas (thomas.greve@haw-kiel.de)
Lecturer(s)	Greve, Thomas (thomas.greve@haw-kiel.de)
Offered in Semester	Wintersemester 2026/27
Module duration	1 Semester
Occurrence frequency	Regular
Module occurrence	In der Regel im Wintersemester
Language	Deutsch
Recommended for international students	No
Can be attended with different study programme	Yes

Curricular relevance (according to examination regulations)
Study Subject: B.Eng. - E - Elektrotechnik (PO 2017, V3) Module type: Wahlmodul Semester: 5
Study Subject: B.Eng. - E - Elektrotechnik (PO 2023, V4) Module type: Wahlmodul Semester: 5
Study Subject: B.Eng. - Me (PO 2024) - Mechatronik (PO 2024, V5) Module type: Wahlmodul Semester: 5
Study Subject: B.Eng. - Wing - Wirtschaftsingenieurwesen - Elektrotechnik (PO 2025, V2) Module type: Wahlmodul Semester: 5
Study Subject: B.Eng. - Wing - Wirtschaftsingenieurwesen - Elektrotechnik (PO 2017, V1) Module type: Wahlmodul Semester: 5
Study Subject: B.Sc. - INF - Informatik (PO 2021,V1) Module type: Wahlmodul Semester: 5

Qualification outcome
Areas of Competence: Knowledge and Understanding; Use, application and generation of knowledge; Communication and cooperation; Scientific self-understanding / professionalism.
Kennenlernen der allgemeinen Konzepte: - Asynchronie in Form von Neben- und Parallelläufigkeit - Multitasking und Multithreading Vermittlung - der C++-Sprachkonstrukte, mit denen diese Konzepte realisiert werden können - von Bibliotheken, für solche Konstrukte, die (noch) nicht im C++-Standard enthalten sind. Die Teilnehmer setzen diese Konstrukte im Rahmen der Programmierübung anhand von Aufgaben ein.

Teilnehmer der Veranstaltung können: - einschätzen, bei welchen Aufgabenstellungen Neben- und Parallelläufigkeit sinnvoll eingesetzt werden kann (und bei welchen nicht) - entscheiden, welche der unterschiedlichen Sprachkonstrukte, die C++ für die Umsetzung bietet, den meisten Nutzen bieten - Neben- und Parallelläufigkeit einschliesslich ggf. erforderlicher Synchronisationsmechanismen in C++ programmieren
Durch die Projektarbeit im Team (2.Teil der Veranstaltung) können die Teilnehmer neben der Umsetzung des Gelernten ihre Fähigkeit trainieren: - nicht triviale softwaretechnische Sachverhalte zu diskutieren und so zu einem gemeinsamen Lösungsansatz für eine gestellte Aufgabe zu kommen - einen effizienten Weg für die Realisierung des Lösungsansatzes zu finden (Aufgabenteilung, Wiederverwendung)
Die Teilnehmer können Aufgabenstellungen, deren Realisierung Neben- oder Parallelläufigkeit voraussetzen, selbstständig identifizieren und lösen

Content information

Content	'The free lunch is over' (Herb Sutter) und die Konsequenzen daraus: Effiziente Nutzung von Multicore-Systemen - Asynchronie in Form von Neben- und Parallelläufigkeit - Prozesse, Threads und Fibers/Coroutinen - Hardware-Threads vs OS-Threads vs Threads of Execution - Synchronisationsmechanismen und deren potentielle Probleme - Tasks vs Threads Umsetzung in C++: - Coroutinen - Möglichkeiten der Darstellung von Parallelläufigkeit: Überladungen von Funktionstemplates aus der Algorithm-Bibliothek des Standards vs std::async vs std::thread - Ergebnisübertragung mit std::promise und std::future - std::packaged_task - Synchronisation durch Semaphoren, Mutexes, Locks, Barriers und Latches - Signalisierte Datenübertragung durch std::condition_variable - atomics Nutzung der boost-Bibliotheken: - boost::process für das Handling von Prozessen und die Interprozesskommunikation - boost::asio::thread_pool für eben diese Ausblick auf Konstrukte die erst mittelfristig im Standard enthalten sein werden: - Executors - Continuation

Literature	--- Allgemeine Aspekte --- The Art of Concurrency Clay Breshears O'Reilly Media, Inc., 2009 ISBN: 978-0-596-52153-0 Multicore-Software Urs Gleim und Tobias Schuele dpunkt.verlag, 2012 ISBN: 978-3-89864-758-8 --- C++ - Spezifika --- The C++ Programming Language, 4th ed. (Chapters 41+42, pp. 1191... 1251) Bjarne Stroustrup Addison-Wesley, 2013 ISBN: 978-0-321-56384-2 C++ Concurrency in Action, 2nd ed. Anthony Williams Manning ISBN: 978-1-617-29469-3 C++ High Performance, 2nd ed. Bjoern Andrist, Viktor Sehr Packt ISBN: 978-1-83921-654-1

Teaching formats of the courses	
Teaching format	**SWS**
Projekt	2
Lehrvortrag + Übung	2

Workload	
Number of SWS	4 SWS
Credits	5,00 Credits
Contact hours	48 Hours
Self study	102 Hours

Module Examination	
Examination prerequisites according to exam regulations	Die Termine mit Anwesenheitspflicht wurden wahrgenommen
BI140 - Portfolioprüfung	Method of Examination: Portfolioprüfung Weighting: 100% wird angerechnet gem. § 11 Absatz 2 PVO: No Graded: Yes Remark: Bestehend aus Praesentation zum Semesterprojekt und abschliessendem Test. Details in der Vorlesung.

Miscellaneous	
Recommended Prerequisites	Bestandenene Modulleistung: Programmieren in C++ (PIC)