Subject card

Subject name and code	Concurrent and parallel programming, PG_00037344
Field of study	Technical Physics

Date of commencement of studies	October 2022	Academic year of realisation of subject	2024/2025
Education level	first-cycle studies	Subject group	Optional subject group Subject group related to scientific research in the field of study
Mode of study	Full-time studies	Mode of delivery	at the university
Year of study	3	Language of instruction	Polish
Semester of study	5	ECTS credits	4.0
Learning profile	general academic profile	Assessment form	assessment

Conducting unit	Institute of Physics and Applied Computer Science -> Faculty of Applied Physics and Mathematics

Name and surname of lecturer (lecturers)	Subject supervisor	dr hab. Jan Franz
	Teachers	dr hab. Jan Franz

Lesson types and methods of instruction	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	15.0	0.0	30.0	0.0	0.0	45
	E-learning hours included: 0.0						

Learning activity and number of study hours	Learning activity	Participation in didactic classes included in study plan	Participation in consultation hours	Self-study	SUM
	Number of study hours	45	10.0	45.0	100

Subject objectives	The aim of the course is to present issues related to the design and implementation of concurrent and parallel computations. In particular, the stages of analysis, design, implementation and evaluation of the correctness of concurrent programs will be discussed.

Learning outcomes	Course outcome	Subject outcome	Method of verification
	K6_K01	The student knows the scenarios where the use of concurrent or parallel programming is necessary to solve the problem.	[SK5] Assessment of ability to solve problems that arise in practice
	K6_W05	The student learns how to solve the classical concurrency problems.	[SW1] Assessment of factual knowledge
	K6_U03	The student practices Java programming.	[SU1] Assessment of task fulfilment

Subject contents	1. Basic Concepts
	1.1 Basic Introduction
	1.2 Thread Concept in Java
	1.3 Elementary Mechanisms for Synchronization
	1.4 Basic Control of Threads
	2. Advanced Concepts
	2.1 Measures for Parallelization
	2.2 Thread Pools
	2.3 Lock Objects
	2.4 Thread-safe Collections
	2.5 Additional Mechanisms for Synchronization
	3. Frameworks for Parallelization
	3.1 Divide-And-Conquer / Fork-Join
	3.2 Parallel Arrays and Streams
	4. Example Applications
	4.1 Monte Carlo Simulation
	4.2 Machine Learning Application
Prerequisites and co-requisites	The fundamentals of Java and/or python programming.

Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	Written test (10 points) and laboratory (30 points). Total 40 points.	50.0%	100.0%

Recommended reading	Basic literature	1. B. Wittman, T. Korb, A. Mathur, Start Concurrent: An Introduction to Problem Solving in Java with a Focus on Concurrency, Purdue University Press, Ashland, Oregon, 2014. 2. T. Rauber, G. Rünger, Parallel Programming: for Multicore and Cluster Systems, Springer Nature, Berlin, 2010. 3. S. Selikoff, J. Boyarsky, OCA/OCP Java SE 8 programmer : practice tests, Sybex, Indianapolis, Indiana, 2017.
	Supplementary literature	1. M. Ben-Ari, "Principles of Concurrent and Distributed Programming", 2nd edition, Addison-Wesley, Upper Saddle River, NJ, 2006. 2. R.-G. Urma, M. Fusco, A. Mycroft, Modern Java in Action, Manning Publications, Shelter Island, 2018. 3. B. Goetz, T. Peierls, J. Bloch, J. Bowbeer, D. Holmes, D. Lea, Java Concurrency in Practice. Addison-Wesley, Upper Saddle River, NJ, 2006. 4. B. J. Evans, J. Clark, M. Verburg, The Well-Grounded Java Developer, Second Edition, Manning Publications, Shelter Island, 2023.
	eResources addresses	Uzupełniające Adresy na platformie eNauczanie: Programowanie współbieżne i równoległe - 2024/25 - Moodle ID: 41175 https://enauczanie.pg.edu.pl/moodle/course/view.php?id=41175
Example issues/ example questions/ tasks being completed	1. Calculate the Speedup of a concurrent program. 2. Convert a sequential program in a concurrent program.	
Work placement	Not applicable	

Document generated electronically. Does not require a seal or signature.