

Subject card

Subject name and code	Parallel programming, PG_00156398						
Field of study	Bioinformatics						
Date of commencement of studies	October 2025		Academic year of realisation of subject		2027/2028		
Education level	Bachelor's studies		Subject group		Optional subject group		
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		3.0		
Learning profile	academic		Assessment form				
Conducting unit	Faculty of Chemistry -> Rector						
Name and surname of lecturer (lecturers)	Subject supervisor		prof. dr hab. Cezary Czaplewski				
	Teachers						
Lesson types	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	0.0	0.0	30.0	0.0	0.0	30
	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	30		0.0		45.0	75
Subject objectives	1. The student acquires the ability to effectively design and conduct parallel computations. 2. To familiarize the student with available parallel programming tools, in particular MPI and multiprocessing libraries. 3. The student acquires basic programming skills using MPI and multiprocessing libraries						
Learning outcomes	Course outcome		Subject outcome		Method of verification		
	[BIOINL3_W01] Has knowledge of computer science technologies, with particular emphasis on programming		The student recognizes and characterizes parallel architectures, parallel libraries, parallel programming tools, and knows MPI and multiprocessing libraries.		[SW2] presentation/project/paper/report		
	[BIOINL3_U01] Graduate is able to program using modern programming tools, including tools dedicated to bioinformatics		The student assesses the usefulness of parallel processing to solve a given problem, runs parallel applications in batch and interactive mode, analyzes source codes using MPI and multiprocessing libraries, creates simple parallel codes using MPI and multiprocessing libraries.		[SU2] presentation/project/paper/report		
Subject contents	Parallel processing as an essential tool in bioinformatics. Types of architectures for parallel computing. Computations in shared and distributed memory systems. Scalability of parallel computing: Amdahl's law. Running parallel tasks on computing clusters, queuing systems. Application of the MPI and multiprocessing libraries in Python programs. Point to point communication: safety and avoiding deadlock. Collective communication. Groups and communicators. Inter-communicators. Data types and user operators in MPI. Virtual topologies. Evaluating the effectiveness of parallelization and profiling parallel programs. Comparison of the MPI and multiprocessing libraries.						

Prerequisites and co-requisites	Prerequisite: • Ability to work in Unix • Python programming skills Formal requirements: • Python with the basics of algorithms • Numerical methods for bioinformaticians • Mathematical methods of bioinformatics		
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	raports	51.0%	50.0%
	project	51.0%	50.0%
Recommended reading	Basic literature	• MPI for Pyton https://mpi4py.readthedocs.io/en/stable/ • W. Gropp, E. Lusk, A. Skjellum, Using MPI. Portable Parallel Programming with the Message-Passing Interface, The MIT Press, Cambridge, 1999. • W. Gropp, E. Lusk, R. Thakur, Using MPI-2. Advanced Features of the Message-Passing Interface. The MIT Press, Cambridge, 1999.	
	Supplementary literature	• I. Foster, Designing and Building Parallel Programs, Addison Wesley, 1995 • M. Snir, S. Otto, S. Huss-Lederman, D. Walker, J. Dongarra, MPI: the Complete Reference, The MIT Press, 1995 • MPI Forum: http://www.mpi-forum.org	
	eResources addresses	Adresy na platformie eNauczanie:	
Example issues/ example questions/ tasks being completed			
Work placement	Not applicable		

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