GDAŃSK UNIVERSITY OF TECHNOLOGY

Subject card

Subject name and code	High Performance Computing Systems, PG_00054279		
Field of study	Informatics		
Date of commencement of studies	February 2026	Academic year of realisation of subject	2025/2026
Education level	second-cycle studies	Subject group	Obligatory subject group in the field of study 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	1	Language of instruction	Polish
Semester of study	1	ECTS credits	3.0
Learning profile	general academic profile	Assessment form	assessment
Conducting unit	Department Of Computer Architecture -> Faculty Of Electronics Telecommunications And Informatics -> Wydziały Politechniki Gdańskiej		
Name and surname of lecturer (lecturers)	Subject supervisor	dr hab. inż. Paweł Czarnul	
	Teachers	mgr inż. Robert Kałaska dr inż. Adam Brzeski dr hab. inż. Paweł Czarnul dr inż. Mariusz Matuszek	

Lesson types and methods of instruction	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	15.0	0.0	15.0	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	10.0	35.0	75

Subject objectives	learning methods and technologies for parallelization and optimization of algorithms in HPC systems

Learning outcomes

Course outcome	Subject outcome	Method of verification
[K7_W08] knows and understands, to an increased extent, the fundamental dilemmas of modern civilisation, the main development trends of scientific disciplines relevant to the field of education	knows how to find applications for parallel computing	[SW1] Assessment of factual knowledge
[K7_U04] can apply knowledge of programming methods and techniques as well as select and apply appropriate programming methods and tools in computer software development or programming devices or controllers using microprocessors or programmable elements or systems specific to the field of study, making assessment and critical analysis of the prepared software as well as a synthesis and creative interpretation of information presented with it	knows how to optimize an implementation of a parallel application	[SU1] Assessment of task fulfilment
[K7_U03] can design, according to required specifications, and make a complex device, facility, system or carry out a process, specific to the field of study, using suitable methods, techniques, tools and materials, following engineering standards and norms, applying technologies specific to the field of study and experience gained in the professional engineering environment	knows how to design and implement a scalable parallel application	[SU1] Assessment of task fulfilment
[K7_U12] is able, to an increased extent, to analyze the operation of components and systems related to the field of study, as well as to measure their parameters and study their technical characteristics, and to plan and carry out experiments related to the field of study, including computer simulations, interpret the obtained results and draw conclusions	knows how to configure a parallel environment, run and profile runs of parallel applications	[SU1] Assessment of task fulfilment

Subject contents

1 Passing criteria 2 Introduction to computing in parallel environments: model of a parallel application: basic parameters of an application and a parallel system, topologies. Assignment criteria. Latency and bandwidth. Values for real systems 3 Parallel architectures: Shared Memory, Distributed Shared Memory, Distributed Memory. 4 Paradigms: master-slave, SPMD, pipelining, divide-and-conquer. Examples 5 MPI: model of an application. Basic API. basic application. 6 MPI: running, various implementations: MPICH, OpenMPI, running on clusters, supercomputers, queueing systems 7 MPI: send communication modes: classic, rsend, bsend, ssend, examples. Non-blocking communication. Blocking communication. Collective vs, point-to-point communication, creation of data types and packing. 8 MPI: communicators 9 MPI: dynamic load balancing: repartitioning, „ghost nodes”. 10 Examples of applications in MPI – performance on TASK clusters (communication modes (SSend,Isend,Rsend,Send), overlapping communication and computations in MPI) 11 MPI: spawning processes dynamically, one-way communication 12 Checkpointing of parallel aplikations 13 Threads vs MPI. OpenMPI 14 Advanced techniques for algorithm parallelization: overlapping communication and communication, other techniques for hiding communication costs 15 OpenMP 16 Parallel I/O in MPI 17 Examples of HPC applications: medical etc. 18 Programming basics in CUDA 19 Programming basics using OpenMP for Xeon Phi

Prerequisites and co-requisites

znajomość języka C, znajomość podstawowych algorytmów sekwencyjnych i struktur danych

Assessment methods and criteria

Subject passing criteria	Passing threshold	Percentage of the final grade
Practical exercise	50.0%	50.0%
Midterm colloquium	50.0%	50.0%

Recommended reading	Basic literature	1 Dokumentacja MPI 2. Dokumentacja OpenMP 3. Dokumentacja CUDA 4. P. Czarnul. Parallel Programming for Modern High Performance Computing Systems. Taylor & Francis. 2018 ISBN 9781138305953
	Supplementary literature	No requirements
	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.