Warsaw University of Technology

COURSE OFFERED IN THE DOCTORAL SCHOOL

Code of the course	4606-ES-0000000-0153	Name of the course	Polish	**Nowoczesne Technologie Obliczeniowe**
			English	Modern Computing Technologies

Type of the course	specialized		
Course coordinator	Gabriel Wlazłowski	Course teacher	Gabriel Wlazłowski, Andrzej Makowski
Implementing unit	Faculty of Physics	Scientific discipline / disciplines*	All disciplines

Level of education	Doctoral studies	Semester	spring		
Language of the course	English				
Type of assessment	credit	Number of hours in a semester	45	ECTS credits	4
Minimum number of participants	10	Maximum number of participants	15	Available for students (BSc, MSc)	Yes/~~No~~

Type of classes		Lecture	Auditory classes	Project classes	Laboratory	Seminar
Number of hours	in a week	1			2	
	in a semester	15			30	

* does not apply to the Researcher's Workshop

1. Prerequisites

- Good knowledge of C language (MANDATORY): in this language examples and code templates will be provided. Basic knowledge of python language.
- Familiar with ssh and sftp protocols (to be able to connect with our cluster and exchange files with it).
- Familiar with Linux system (editing files, managing files, executing commands…).

2. Course objectives

Student / PhD students are introduced to the practical aspects of using computing clusters and supercomputers. The student / doctoral student acquires the ability to design and create programs that solve typical computational problems using parallel computing and graphics processors during the course. The numerical problems are related to linear algebra, spectral analysis, and solving partial differential equations. The subject consists of a lecture and a laboratory part. To carry out laboratory tasks, students will receive access to the computing cluster of the Faculty of Physics DWARF. Students will also learn about polish computational resources and methods of accessing them.

3. Course content (separate for each type of classes)

Lecture

1. Fundamentals of the architecture of computing systems. Computing clusters. Supercomputers.
2. Definition of computing performance and computation intensity. Parallel computing models: multithreaded and multiprocess computing.
3. Parallel computation with the shared memory model. OpenMP specification.
4. Parallel computing with distributed memory. MPI specification.
5. Performance-limited and bandwidth-limited codes. Combining high-level with low-level

languages (C & Python as an example). 6. Signal processing, FFTW library. 7. Lineral algebra. BLAS and LAPACK libraries. 8. Parallel linear algebra. BLACS, PBLAS, and ScaLAPACK libraries. 9. Parallel reading and writing files, lustre filesystems. MPI I/O specification. 10. Accelerated computing with graphical processing units (GPUs), CUDA and HIP languages. MAGMA library. 11. Processing of space matrices. Solving of partial differential equations by reducing it to problems that involve sparse matrices. 12. Scaling of the numeric code. Weak and strong scaling. 13. Computing resources in Poland and Europe and methods of accessing them.
Laboratory
1. Students will have access to the computing cluster of the Faculty of Physics DWARF. 2. During the laboratories, the students will create a C-based computational program based on the instructor's instructions and using provided code templates. The parallel / heterogeneous programming methods presented in the lecture will be used. The prepared program will have to be run on a suitable computer, and basic scalability tests will be performed. The results of these tests will be used to create a short report. 3. In total students will create about 10 codes that demonstrate the usage of various HPC techniques/libraries. 4. For laboratory purposes, students will have to configure their own computers, which they will use as terminals to work with the DWARF cluster.

4. Learning outcomes			
Type of learning outcomes	Learning outcomes description	Reference to the learning outcomes of the WUT DS	Learning outcomes verification methods*
Knowledge			
K01	knowledge latest developments in high-performance computing	SD_W3	report evaluation
K02	knowledge of standards numerical libraries used for scientific computation	SD_W3	report evaluation
Skills			
S01	he/she can write numerical code that utilizes supercomputers	SD_U1	report evaluation
S02	he/she can solve typical problems from the scientific computation domain, like signal processing, solving algebraic and partial differential equations	SD_U1	report evaluation
S02	he/she can write a proposal for access to (super)computing resources in Poland and Europe	SD_U4	report evaluation
Social competences			
SC01	he/she understands the role of high-performance computing (HPC) in modern science and the process of migrating technical	SD_K2	report evaluation

	solutions from the HPC domain to general use		

*Allowed learning outcomes verification methods: exam; oral exam; written test; oral test; project evaluation; report evaluation; presentation evaluation; active participation during classes; homework; tests

5. Assessment criteria
• Selected by the teacher codes (created during labs) will be graded. • The teacher will set for each problem a deadline for submission of the codes together with reports presenting the results. • Selected by the teacher codes will be graded from 3.0 to 5.0. • The final grade is computed as a weighted average (for each code weight will be assigned according to the complexity of the problem). • To pass the course, the student must get at least 3.0 from each task.

6. Literature
1.Thomas Rauber, Gudula Rünger, „Parallel Programming for Multicore and Cluster Systems", Springer 2013,
2. R.W. Hamming, "Numerical Methods for Scientists and Engineers", Dover Publications, 1987
3. Foster I., "Designing and Building Parallel Programs", e-book: www-unix.mcs.anl.gov/dbpp;
4.. Strony internetowe: www.openmp.org, www.mpi-forum.org, www.netlib.org, https://developer.nvidia.com/cuda-zone.

7. PhD student's workload necessary to achieve the learning outcomes**		
No.	Description	Number of hours
1	Hours of scheduled instruction given by the academic teacher in the classroom	45
2	Hours of consultations with the academic teacher, exams, tests, etc.	20
3	Amount of time devoted to the preparation for classes, preparation of presentations, reports, projects, homework	30
4	Amount of time devoted to the preparation for exams, test, assessments	20
	Total number of hours	**115**
	ECTS credits	**4**

** 1 ECTS = 25-30 hours of the PhD students work (2 ECTS = 60 hours; 4 ECTS = 110 hours, etc.)

8. Additional information	
Number of ECTS credits for classes requiring direct participation of academic teachers	4
Number of ECTS credits earned by a student in a practical course	4