Warsaw University of Technology | Doctoral School No. 1

Course offered in the Doctoral School No. 1 – Spring semester of the 2021/2022 academic year

TITLE
Modern Computing Technologies
CONDUCTING UNIT
Doctoral School No. 1
SCIENTIFIC DISCIPLINE
Physical science
IMPLEMENTING UNIT
105000 - Faculty of Physics
SUMMARY DESCRIPTION
Student / PhD students are introduced to the practical aspects of the use of 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, computation of derivatives. The subject consists of a lecture and laboratory part. To carry out laboratory tasks, students will receive access to the computing cluster of the Faculty of Physics DWARF. Students will also get knowledge about polish computational resources and methods of accessing them.
FULL DESCRIPTION
Lecture content: 1. Fundamentals of the architecture of computing systems. Memory models. Introducing the concept of computing accelerators (e.g. graphics processors). 2. Idea and models of parallel computing, concepts of multithreaded and multiprocess computing. 3. Parallel computations in the shared memory model. OpenMP specification. 4. Parallel computing with distributed memory. MPI specification. 5. Heterogeneous computing, the use of CUDA technology. 6. Review of basic methods and algorithms used in scientific simulations, signal and data analysis (discrete Fourier transform, matrix diagonalization, systems of linear equations, integration, Monte Carlo methods, data searching). 7. Implementation of algorithms, numerical libraries for FFT algorithms, linear algebra (BLAS, LAPACK), array reduction, use of sparse matrices.

8. Comparison of programming solutions to computational problems mentioned in point 6 in shared / distributed memory models and in heterogeneous computing.
9. Computing clusters and supercomputers. Scaling of the numeric code, strong and weak scaling.
10. Combining low and high level languages: C+python will be demonstrated as an example.
11. Review of available computing systems in Poland and abroad and methods of accessing them.

Laboratory:
1. In order to perform laboratory tasks, students will have access to the computing cluster of the Faculty of Physics DWARF.
2. For laboratory purposes, students will have to configure their own computers, which they will use as terminals to work with the DWARF cluster.
3. During the laboratories, the student's task will have to create a computational program in C based on the instructor's instructions or with the use of provided code templates by the teacher. The methods of parallel / heterogeneous programming 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.

Requirements :
* you can write programs C language (MANDATORY): in this language examples and code templates will be provided.
* you are familiar with ssh and sftp protocols (to be able to connect with our cluster and exchange files with it);
* you are familiar with the basics of Linux (copying files, executing commands...);

LITERATURE

Thomas Rauber, Gudula Rünger, „Parallel Programming for Multicore and Cluster Systems", Springer 2013,
Foster I., "Designing and Building Parallel Programs", e-book: www-unix.mcs.anl.gov/dbpp;
Strony internetowe: www.openmp.org, www.mpi-forum.org, www.netlib.org, https://developer.nvidia.com/cuda-zone;

LEARNING OUTCOMES

During the course, the student / doctoral student acquires the ability to design and create programs that solve typical scientific problems using parallel computing and graphics processors.

ASSESSMENT METHODS AND CITERIA; COURSE COMPLETION FORM

Students must create 10 programs in C and demonstrate that they work.
The programs are created during the laboratories.
The teacher will set for each problem a deadline for submission of the codes together with short reports presenting the results.
Each task is graded from 3.0 to 5.0.
The final grade is computed as a weighted average (for each task 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.

LANGUAGE OF THE COURSE		ECTS CREDITS
English		5
TYPE OF CLASSES	**NUMBER OF HOURS**	**COURSE INSTRUCTOR**
Lecture	15	Gabriel Wlazłowski, dr hab. inż.
Laboratory	30	Andrzej Makowski, mgr inż.