UNIVERSIDADE DA CORUÑA

Teaching Guide				
Identifying Data				2020/21
Subject (*)	Advanced Parallel Programming		Code	614473107
Study programme	Mestrado Universitario en Computación de Altas Prestacións / High Performance Computing (Mod. Presencial)			
Descriptors				
Cycle	Period	Year	Type	Credits
Official Master's Degree	2nd four-month period	First	Optional	6
Language	SpanishGalicianEnglish			
Teaching method	Hybrid			
Prerequisites				
Department	Departamento profesorado másterEnxeñaría de Computadores			
Coordinador	Fraguela Rodriguez, Basilio Bernardo	E-mail	basilio.fraguela@udc.es	
Lecturers	Darriba López, Diego Fraguela Rodriguez, Basilio Bernardo	E-mail	diego.darriba@udc.es basilio.fraguela@udc.es	
Web	aula.cesga.es			
General description	This subject will increase the knowledge on parallel programming acquired by the students in the previous quarter in subjects such as "Parallel Programming" and "Programming of heterogeneous architectures". The aim will be that the students learn to optimize parallel codes for big parallel architectures or current supercomputers, using for their tests the resources provided by the Centro de Supercomputación de Galicia (CESGA) and the Group of Architecture of Computers (GAC) of the Universidade da Coruña (UDC). We will focus on those aspects of the parallel applications that usually penalize performance, such as the communications, load unbalance, memory access patterns or the management of I/O. We will also tackle multiplatform computing, which allows to take advantage of the task level parallelism by using several hardware accelerators, as well as hybrid computing, where the same application uses several parallel programming paradigms in order to obtain good performance in clusters with multi-core computers and/or hardware accelerators.			

Contingency plan	1. Modifications to the contents There will be no modifications to the contents. 2. Methodologies *Teaching methodologies that are maintained - Laboratory practice - Supervised projects - Guest lecture / keynote speech *Teaching methodologies that are modified None 3. Mechanisms for personalized attention to students - Slack or Microsoft Teams channel of the subject: at any time during the semester - email: at any time during the semester - Cesga Classroom: During the entire semester for the supply of material and collection of works - Face-to-face and distance communication by jitsi or Microsoft Teams: During both face-to-face and remote classes, both in the master sessions and in laboratory practices 4. Modifications in the evaluation None. The supervised projects will continue to weigh 100%, valued in exactly the same way. *Evaluation observations: None 5. Modifications to the bibliography or webgraphy None

Study programme competences	
Code	Study programme competences
A1	CE1 - Define, evaluate and select the most appropriate architecture and software to solve a problem
A2	CE2 - Analyze and improve the performance of a given architecture or software
A4	CE4 - Deepen in the knowledge of different programming tools and programming languages in the field of the high performance computing
A5	CE5 - Analyze, design and implement efficient parallel algorithms and applications
A7	CE7 - Know the emerging technologies in the supercomputing field
B1	CB6 - Possess and understand the knowledge that give a baseline or opportunity to be original in the development and/or application of ideas, often in a research environment
B2	CB7 - The students have to know how to apply the acquired knowledge and their capacity to solve problems in new or hardly explored environment inside wider contexts (or multidiscipinary) related to its area of development

B5	CB10 - The students have to possess learning skills that allows them to continue to study in a mainly self-driven or autonomous manner
B6	CG1 - Be able to search and select useful information to solve complex problems, using the bibliographic sources of the field
B9	CG4 - Be able to plan and do research, development and innovation tasks in high performance computing related environments
C1	CT1 - Use the basic technologies of the information and computing technology field required for the professional development and the long-life learning

Learning outcomes			
Learning outcomes	Study programme competences		
Know advanced techniques for the optimization of parallel codes	AJ1 AJ2 AJ5	BJ1 BJ2 BJ5 BJ6 BJ9	
Control the affinity and load balance of tasks	AJ5	BJ1 BJ2 BJ5 BJ6 BJ9	CJ1
Optimize communications in distributed memory systems	AJ2 AJ4 AJ5	BJ1 BJ2 BJ5 BJ6 BJ9	CJ1
Perform parallel input/output operations	AJ4 AJ5	BJ1 BJ2 BJ5 BJ6 BJ9	CJ1
Program systems with several hardware accelerators	AJ4 AJ5 AJ7	BJ1 BJ2 BJ5 BJ6 BJ9	CJ1
Program systems with shared/distributed memory	AJ4 AJ5 AJ7	BJ1 BJ2 BJ5 BJ6 BJ9	CJ1

Contents	
Topic	Sub-topic
1- Advanced techniques for the optimization of parallel codes	-
2- Affinity control and load balance	-
3- Optimization of communications in distributed memory systems	-
4- Parallel input/output	-
5- Hybrid programming for systems with several hardware accelerators	-

6- Hybrid programming for systems with shared/distributed memory	-

Planning				
Methodologies / tests	Competencies	Ordinary class hours	Student?s personal work hours	Total hours
Laboratory practice	A2 A5 C1	21	63	84
Supervised projects	A1 A2 A4 A5 A7 B1 B2 B5 B6 B9 C1	0	45	45
Guest lecture / keynote speech	A1 A4 A7 B1	20	0	20
Personalized attention		1	0	1
(*)The information in the planning table is for guidance only and does not take into account the heterogeneity of the students.				

Methodologies	
Methodologies	Description
Laboratory practice	In these classes, directed tasks are carried out that allow the student to become familiar from a practical point of view with the contents exposed in the theoretical classes.
Supervised projects	They consist in the development of projects in which the student has to use the acquired knowledge to solve different problems in an autonomous way.
Guest lecture / keynote speech	Theoretical classes, in which the content of each topic is exposed. The student will have all the necessary material before the class and the teacher will promote an active attitude, asking questions that clarify specific aspects and leaving open questions for the student's reflection.

Personalized attention	
Methodologies	Description
Supervised projects Laboratory practice	Both in the lab practices as well as during the development of the supervised projects, the students will be able to present questions, doubts, etc. The teacher, taking care of these requests, will review concepts, solve new problems or use any activity that considers appropriate to resolve the issues raised.

Assessment			
Methodologies	Competencies	Description	**Qualification**
Supervised projects	A1 A2 A4 A5 A7 B1 B2 B5 B6 B9 C1	Quality of the work developed and progress of the student during its completion	100

Assessment comments
In the activities of distance evaluation students may be required to apply mechanisms that guarantee their identity as well as the authorship of the evaluable elements presented. All the evaluation activities included in this guide conform the process of continuous evaluation of the subject. Neither the classes nor the evaluation activities require the student's presence. This, together with the fact that all the materials of the subject are available in the education web platform of the degree, favors the work and the evaluation of the students enrolled part-time and with academic allowance of teaching exemption.

Sources of information

Basic	-Â Using Advanced MPI: Modern Features of the Message-Passing Interface. 2014. W. Gropp, T. Hoefler, R. Thakur, E. Lusk. MIT Press-Â Using OpenMP: The Next Step: Affinity, Accelerators, Tasking, and SIMD (Scientific and Engineering Computation). 2017. R. van der Pas, E. Stotzer, C. TerbovenÂ . MIT Press-Â OpenCL Programming Guide. 2011. A. Munshi, B. Gaster, T. G. Mattson, J. Fung, D. Ginsburg.Â Addison-Wesley/Pearson Education- Using Advanced MPI: Modern Features of the Message-Passing Interface. 2014. W. Gropp, T. Hoefler, R. Thakur, E. Lusk. MIT Press- Using OpenMP: The Next Step: Affinity, Accelerators, Tasking, and SIMD (Scientific and Engineering Computation). 2017. R. van der Pas, E. Stotzer, C. Terboven . MIT Press- OpenCL Programming Guide. 2011. A. Munshi, B. Gaster, T. G. Mattson, J. Fung, D. Ginsburg. Addison-Wesley/Pearson Education
Complementary	- Multi-core programming. 2006. S. Akhter e J. Roberts. Intel Press.Â - Professional CUDA C Programming. 2014. J. Cheng, M. Grossman, T. McKercher. Wross.- Multi-core programming. 2006. S. Akhter e J. Roberts. Intel Press. - Professional CUDA C Programming. 2014. J. Cheng, M. Grossman, T. McKercher. Wross.

Recommendations
Subjects that it is recommended to have taken before
Parallel Programming/614473102 Heterogeneous Programming/614473103
Subjects that are recommended to be taken simultaneously
Subjects that continue the syllabus
Master's Thesis/614473111
Other comments
Due to the strong interrelation between the theoretical part and the practical part, and the progressiveness in the presentation of concepts closely related to each other in the theoretical part, it is advisable to dedicate a time of study or daily review.In this subject, intensive use of online communication tools will be made: videoconference, email, chat, etc.