

Teaching Guide					
	Identifying Data			2020/21	
Subject (*)	HPC Tools Code			614473105	
Study programme	Mestrado Universitario en Compo	utación de Altas	Prestacións / Hig	gh Performance Computir	g (Mod. Presencial)
		Descri	ptors		
Cycle	Period	Yea	ar	Туре	Credits
Official Master's Degree	e 1st four-month period	Firs	st	Optional	6
Language	English				
Teaching method	Hybrid				
Prerequisites					
Department	Enxeñaría de Computadores				
Coordinador	Padron Gonzalez, Emilio Jose		E-mail	emilio.padron@ud	lc.es
Lecturers	Andrade Canosa, Diego		E-mail	diego.andrade@u	dc.es
	Padron Gonzalez, Emilio Jose			emilio.padron@ud	lc.es
Web	aula.cesga.es				
General description	The objective of this course is to	get the students	familiar with the	most common types of a	oplication that are candidates to
	use HPC, besides being introduced to the main tools and implementations existing for them, understanding the challenges			, understanding the challenges	
	to be addressed for their parallelization and performance tuning. All this will allow the students to obtain a general			nts to obtain a general	
	knowledge about the HPC field a	ind its different a	applications and u	ise cases.	
	Furthermore, the students will lea	arn what tools ca	an be used to car	ry out the performance ch	aracterization and
	benchmarking tasks in HPC environments, and how these tools can be leveraged to drive the parallelization and				
	performance tuning of an application on a specific platform. This will allow the students to be able to analyze the expected				
	performance on that system, ider	ntifying the differ	ent hot spots and	d focussing the optimization	on efforts on them.
	Finally, the students will learn diff	ferent technolog	ical alternatives f	or a fast and efficient dep	loyment of HPC applications.
This will allow them to be able to easily and effectively deliver and execute HPC applications in different environments.					



Contingency plan	1. Modifications to the contents
	- None
	2. Methodologies
	*Teaching methodologies that are maintained
	- The teaching methodologies used in this course are maintained, but changing the teaching method from "hybrid" to
	"Non-attendance".
	*Teaching methodologies that are modified
	- None, only the teaching method is modified: hybrid -> non-attendance
	3. Mechanisms for personalized attention to students
	- The previously planned but limiting communication channels to e-mail and the UDC's Teams tool
	4. Modifications in the evaluation
	- None, the evaluation is already online in this course
	*Evaluation observations:
	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
A3	CE3 - Know the high performance computing basic concepts
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
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
B3	CB8 - The students have to be able to integrate knowledge and face the complexity to make judgments from information, despite being
	partial and limited, includes reflexions about the social and ethical responsabilities linked to the application of their judgements and
	knowledge
B4	CB9 - The students have to be able to communicate their conclusions, their knowledge and the reasons that hold them to specialized and
	non specialized audience in a clear and unambiguous manner
B6	CG1 - Be able to search and select useful information to solve complex problems, using the bibliographic sources of the field
B8	CG3 - Be able to maintain and extend properly funded theoretical hypothesis to allow the introduction and exploitation of novel and
	advanced technologies in 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



C4	CT4 - Value the importance of research, innovation and the technological development in the socioeconomical and	cultural a	advance	of the
	society			
	Learning outcomes			
	Learning outcomes	Study	/ progra	mme
		cor	npetend	es
Students w	ill know the most common types of applications in which HPC techniques are usually applied.	AJ1	BJ1	CJ1
		AJ2	BJ6	
Students w	ill learn to use tools to characterize and represent the performance of applications.	AJ3	BJ3	CJ4
		AJ4	BJ9	
Students w	ill learn to use tools to compile, generate and deploy software in HPC environments.	AJ3	BJ1	CJ1
		AJ5	BJ4	
			BJ8	

	Contents
Торіс	Sub-topic
A survey of main application types in HPC. For each type	1. Problem: formal description.
we?ll see:	2. Parallelization and performance tuning challenges.
	3. Existing approaches.
Tools to measure, characterize and represent the	1. Usage of performance characterization and benchmarking tools, such as software
performance of HPC applications.	monitoring and hardware counters.
	2. Hot spot detection to drive the optimization process.
	3. Application of performance models to this process.
	4. Tools for application performance representation.
Tools for the compilation, generation and deployment of HPC	1. Code compilation, optimization and generation in a compiler.
software.	2. Code optimization with a compiler.
	3. Automatic parallelization and vectorization.
	4. Software development tools.
	5. Leveraging containers for the easy deployment of HPC applications.

Planning				
Methodologies / tests	Competencies	Ordinary class	Student?s personal	Total hours
		hours	work hours	
Guest lecture / keynote speech	A3 B1 C4	23	0	23
Laboratory practice	A1 A2 A4 A5 C1	18	52	70
Supervised projects	B3 B4 B6 B8 B9	0	54	54
Mixed objective/subjective test	B4 B6	2	0	2
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
Guest lecture /	Lectures, discussing the different lessons of the course. Students will have available all the necessary material in advance and
keynote speech	the teacher will promote an active attitude in the classroom, asking questions that may clarify specific aspects and leaving
	open issues for student reflection.
Laboratory practice	Lab sessions, allowing the students to become familiar from a practical standpoint with the issues discussed in the lectures.
Supervised projects	Guided task fulfillment: students apply the acquired knowledge to solve different problems autonomously.
Mixed	Written test/exam to show that the students have acquired the Degree's competences trained in this course by answering
objective/subjective	theoretical questions and solving exercises.
test	



	Personalized attention
Methodologies	Description
Laboratory practice	Personalized attention is guaranteed during the development of the laboratory practices and supervised projects, being
Supervised projects	essential to guide students in the fulfillment of their tasks. This personalized attention is also useful to validate and evaluate
	the work carried out throughout the different development stages, until finished.
	Furthermore, it is recommended for students to leverage the teacher's office hours as a complementary assistance tool.

Assessment			
Methodologies	Competencies	Description	Qualification
Mixed	B4 B6	Written test/exam to show that the students have acquired the Degree's competences	30
objective/subjective		trained in this course by answering theoretical questions and solving exercises.	
test			
Supervised projects	B3 B4 B6 B8 B9	Guided task fulfillment: students apply the acquired knowledge to solve different	70
		problems autonomously.	

Asses	sment comments

	Sources of information
Basic	[1] Computer Architecture: A Quantitative Approach (5th or 6th Ed.). John L. Hennessy, David A. Patterson. Morgan
	Kaufmann. ISBN 978-0123838728 (5th Ed. 2011) 978-0128119051 (6th Ed. 2017)[2] Performance Tuning of Scientific
	Applications. David H. Bailey, Robert F. Lucas, Samuel Williams. CRC Press. ISBN 978-1439815694[1] Computer
	Architecture: A Quantitative Approach (5th or 6th Ed.). John L. Hennessy, David A. Patterson. Morgan Kaufmann.
	ISBN 978-0123838728 (5th Ed. 2011) 978-0128119051 (6th Ed. 2017)[2] Performance Tuning of Scientific
	Applications. David H. Bailey, Robert F. Lucas, Samuel Williams. CRC Press. ISBN 978-1439815694
Complementary	[3] Intel® C++ Compiler Developer Guide and Reference
	https://software.intel.com/cpp-compiler-developer-guide-and-reference[4] A Guide to Vectorization with Intel® C++
	Compilers https://software.intel.com/sites/default/files/m/4/8/8/2/a/31848-CompilerAutovectorizationGuide.pdf[5] Intel®
	VTune? Amplifier Help https://software.intel.com/en-us/vtune-amplifier-help[6] Free Software Foundation, Inc.: Using
	the GNU Compiler Collection (GCC). https://gcc.gnu.org/onlinedocs

Recommendations
Subjects that it is recommended to have taken before
Parallel Programming/614473102
Subjects that are recommended to be taken simultaneously
Subjects that continue the syllabus
Other comments
Because of the strong interrelation between the lectures and the lab
sessions, and the progressive presentation of concepts very related each
other in the lectures, it is recommended to dedicate enough time to a
daily study or review. This course will leverage online communication tools in quite an intensive way: videoconference, e-mail, chat, etc.



(\*)The teaching guide is the document in which the URV publishes the information about all its courses. It is a public document and cannot be modified. Only in exceptional cases can it be revised by the competent agent or duly revised so that it is in line with current legislation.