| Teaching Guide | | | | |
|---|---|---|---|---|
| Identifying Data | | | | 2020/21 |
| Subject (*) | High Performance Architecture | | Code | 614473101 |
| Study programme | Mestrado Universitario en Computación de Altas Prestacións / High Performance Computing (Mod. Presencial) | | | |
| Descriptors | | | | |
| Cycle | Period | Year | Type | Credits |
| Official Master's Degree | 1st four-month period | First | Obligatory | 6 |
| Language | SpanishEnglish | | | |
| Teaching method | Hybrid | | | |
| Prerequisites | | | | |
| Department | Departamento profesorado másterEnxeñaría de Computadores | | | |
| Coordinador | Doallo Biempica, Ramon | E-mail | ramon.doallo@udc.es | |
| Lecturers | Andrade Canosa, Diego<br>Doallo Biempica, Ramon<br>Touriño Dominguez, Juan | E-mail | diego.andrade@udc.es<br>ramon.doallo@udc.es<br>juan.tourino@udc.es | |
| Web | aula.cesga.es | | | |
| General description | In this course, the students complete their knowledge about HPC architectures, to this end, we consider modern parallel architectures both from the functional point of view to their design. Thi knowledge will facilitate the student to design correct and efficient parallel algorithms based on the architectural characteristics of the target systems. Courses related with the programming will benefited from this one. | | | |
| Contingency plan | 1. Modifications to the contents<br><br>None<br><br>2. Methodologies<br>*Teaching methodologies that are maintained<br><br>All<br><br>*Teaching methodologies that are modified<br><br>None<br><br>3. Mechanisms for personalized attention to students<br><br>Using the teams platform<br><br>4. Modifications in the evaluation<br><br>The evaluation procedure is already suitable for distance teaching.<br><br>*Evaluation observations:<br><br>5. Modifications to the bibliography or webgraphy | | | |

| Study programme competences / results | |
|---|---|
| Code | Study programme competences / results |
| 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 |
| A8 | CE8 - Be able to apply the acquired knowledge, capabilities and aptitudes to the profesional environment, planning, managing and evaluating project in the high performance computing 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 |
| 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 |
| 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 |
| B7 | CG2 - Elaborate adqueately and originally written essays or motivated reasonings, write planings, work projects, scientific papers and formulate reasonable hypothesis |
| B9 | CG4 - Be able to plan and do research, development and innovation tasks in high performance computing related environments |
| B10 | CG5 - Be able to work in teams, specially multidisciplinary, and do a proper time and people management and decision taking |
| 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 / results | | |
| The student will know the different types of parallel architectures and their classification. | AJ1<br>AJ3 | BJ1<br>BJ5 | CJ1 |
| The student will study the basics about organization and design of a parallel architecture, both at microarchitecture level and multiprocessor systems level. | AJ2<br>AJ8 | BJ2<br>BJ4<br>BJ6 | |
| The student will know the design principles an main componentes of a multiprocessor system. | AJ2<br>AJ3<br>AJ8 | BJ1<br>BJ3<br>BJ7<br>BJ9<br>BJ10 | CJ1 |
| The student will learn to analyse parallel architecture performance. | AJ2<br>AJ4<br>AJ8 | BJ4<br>BJ7<br>BJ9 | CJ1 |

| Contents | |
|---|---|
| Topic | Sub-topic |
| Chapter 1. Parallel computers | - Historic introduction<br>- Levels of parallelism: form microarchitecture to supercomputers<br>- Classification |
| Chapter 2. Design of multiprocessors, multicores and manycores | - Introduction<br>- Architecture of multiprocessors, multicores and manycores<br>- Memory architecture |

| | |
|---|---|
| Chapter 3. Cache Coherence | - Protocols<br>- Snooping (UMA systems)<br>- Protocols based on directories (CC-NUMA systems) |
| Tema 4. Sincronización e consistencia de memoria en multiprocesadores | - Primitivas de sincronización<br>- Soporte hardware para sincronización<br>- Implementaciones software de sincronización<br>- Modelos de consistencia de memoria<br>- Comparación entre os modelos de consistencia |
| Chapter 5. Interconexion networks | - Types of networks<br>- Main components<br>- Performance<br>- Design |
| Chapter 6. Distributed systems: clusters | - Introduction<br>- Cluster architecture<br>- Nodes<br>- Interconnection networks<br>- Software<br>- Tools<br>- Applications<br>- Load balance |
| Chapter 7. Introduction to performance analysis. | - Motivation<br>- Basic concepts<br>- Characterization of performance issues<br>- Architecture features related to performance |

| Planning | | | | |
|---|---|---|---|---|
| Methodologies / tests | Competencies / Results | Teaching hours (in-person & virtual) | Student?s personal work hours | Total hours |
| Guest lecture / keynote speech | A1 A3 B1 B5 | 22 | 0 | 22 |
| Laboratory practice | A2 A4 B2 B6 B10 C1 | 24 | 24 | 48 |
| Supervised projects | A8 B3 B4 B7 B9 | 0 | 72 | 72 |
| Mixed objective/subjective test | B4 B7 | 2 | 0 | 2 |
| Personalized attention | | 6 | 0 | 6 |
| (*)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 / keynote speech | The lecturer presents contents of the subject, and asks questions to the student in order to improve learning. There can also be discussions about specific topics. |
| Laboratory practice | Practices and exercices are done in laboratory to support contents explained at keynote speech. |
| Supervised projects | Students will develop individually or joined to other students specific projects/works. It could be possible to present to the rest to the students these works. |
| Mixed objective/subjective test | Some questions about practice and supervised projects can be done by lecturer. |

| Personalized attention | |
|---|---|
| Methodologies | Description |

| | |
|---|---|
| Laboratory practice<br>Supervised projects | Laboratory practice:<br>Lecturer and student analyse the practices done by the student.<br><br>Supervised projects:<br>Students receive lecturer guidance about their assigned supervised projects, and the acomplishment of the scheduled goals are verified periodically. |

| Assessment | | | |
|---|---|---|---|
| Methodologies | Competencies / Results | Description | **Qualification** |
| Laboratory practice | A2 A4 B2 B6 B10 C1 | Valórase o correcto funcionamento, a estructuración do código, e aa comprensión dos conceptos traballados. Tamén valórase a participación activa do estudante durante as sesións de prácticas. | 39 |
| Supervised projects | A8 B3 B4 B7 B9 | No caso de desenvolvemento de código, valoranse os mesmos aspectos que nas prácticas. No caso de traballos escritos, valorase a capacidade de comprensión e síntesis sobre o tema proposto, e a calidade da presentación. | 59 |
| Mixed objective/subjective test | B4 B7 | Tanto no caso das prácticas como dos traballos tutelados o profesor pode facer preguntas concretas aos estudantes que poden complementar a avaliación. | 2 |

| Assessment comments |
|---|
| Evaluation is done in a continuous way based on the supervised projects delivered by the students (60%), and practices and active participation of the students (40%)<br>The student can be requested to identify themselves by an official identification document in the evaluation process. |

| Sources of information |
|---|

| | |
|---|---|
| **Basic** | Dado que se tratan de reflectir non soamente os fundamentos da arquitectura de supercomputadores senón tamén os avances máis recentes, moita da información bibliográfica consultarase en artigos publicados en revista e dispoñibles online e noutras fontes de consulta dispoñibles online. A bibliografía básica necesaria para seguir cada parte da materia a irá indicando o profesor durante as clases. Bibliografía básica. Os libros polos que se segue máis directamente partes da materia son:1. Arquitectura de Computadores, Xullo Ortega, Mancia Anguita e Alberto Prieto. Thompson. 2005.2. High Performance Cluster Computing, Rajkumar Buyya, ed., Prentice Hall PTR, 1999. ISBN 0-13-013784-7, 0-13-013785-5. |
| **Complementary** | Bibliografía complementaria. Os seguintes son libros que permiten consultar máis en profundidade algúns contidos:1. Parallel Computer Architecture, David E. Culler, Jaswinder Pal Singh e Anoop Gupta. Morgan Kaufmann Publishers. 1999.2. In Search of Clusters, 2ª ed., Gregory Pfister, Prentice Hall, 1998, ISBN: 0138997090.3. Organización e Arquitectura de Computadores (7ª edición), W. Stallings. Prentice Hall. 2007.4. Computer Architecture: a Quantitative Approach (6ª edición), John L. Hennessy e David A. Patterson. Morgan Kaufmann Publishers. 2017. |

| **Recommendations** |
|---|
| **Subjects that it is recommended to have taken before** |
| |
| **Subjects that are recommended to be taken simultaneously** |
| Parallel Programming/614473102 |
| **Subjects that continue the syllabus** |
| Heterogeneous Programming/614473103<br>HPC on the Cloud/614473106<br>Advanced Parallel Programming/614473107 |
| **Other comments** |
| |

**(*)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.**