HIGH PERFORMANCE COMPUTING

Grado en Computación e Inteligencia Artificial / Bachelor in Computer Science and Artificial Intelligence BCSAI SEP-2025 HPC-CSAI.3.M.A

Area Computer Science
Number of sessions: 30
Academic year: 25-26
Degree course: THIRD
Number of credits: 6.0
Semester: 1º
Category: COMPULSORY
Language: English

Professor: **OSCAR DIEZ GONZALEZ**

E-mail: odiezg@faculty.ie.edu

Dr. Oscar Diez is the head of quantum computing (HPC and Quantum Technologies Unit) at European Commission (EC), the executive branch of the European Union (EU). Earlier he was involved in the setup of the EuroHPC Joint Undertaking and the procurement of the EU Supercomputers (LUMI, Leonardo, Marenostrum V) that are now among the top supercomputers in the World. Previously, he was Head of Sector in DIGIT (EC) and Head of Datacentre at the European Medicines Agency (EMA) in London.

He holds a PhD in Computer Science from Universidad Politécnica in Madrid (UPM), a degree in Computer science from UPM, a Master's degree from Universidad Pontificia de Salamanca and a Master's degree in Open eGovernment from the University of Stockholm. He has different IT certifications (TOGAF, PMI, PRINCE2, CISSP, CISA, CISM, TOGAf 9, IVIL v3 expert, ISO27001 ISMS).

Office Hours

Office hours will be on request. Please contact at:

odiezg@faculty.ie.edu

SUBJECT DESCRIPTION

This course presents an introduction of high-performance computing (HPC) with a modern perspective, focusing on parallel coding, distributed systems, and applications in machine learning and deep learning. It introduces students to the evolution and fundamental concepts of HPC, including architectural designs of HPC systems, resource management, and performance optimization techniques. Emphasizing practical applications, the course explores the integration of HPC in solving complex problems in health, neurosciences, and various scientific disciplines using parallel computing essentials like MPI and OpenMP, alongside introductions to GPU computing and cloud-based HPC solutions.

The curriculum is designed to equip students with the skills necessary for developing efficient HPC applications, with a significant focus on performance optimization, parallel algorithms, and scalability issues. Special attention is given to the emerging fields of distributed machine learning and deep learning at scale, preparing students to tackle current and future challenges in artificial intelligence with high-performance computing solutions.

Through a combination of theoretical lectures and hands-on practical sessions, students will learn to navigate the complexities of parallel programming, optimize computational fluid dynamics simulations, and implement distributed machine learning models. The course culminates in a comprehensive overview of the frontiers of HPC, including neuromorphic and quantum computing, setting the stage for exploring post-exascale computing paradigms.

By the end of the course, students will have a broad knowledge of HPC systems, their applications across various domains, and the competencies to utilize parallelism for scalability and efficiency. Whether aiming to develop advanced hardware or software systems, use HPC tools in scientific research, or manage and maintain HPC infrastructure, students will be well-prepared to contribute significantly to their chosen fields.

LEARNING OBJECTIVES

This course is structured into six modules, encompassing both the theoretical underpinnings and practical applications of High Performance Computing (HPC), with an emphasis on parallel computing, distributed systems, and machine and deep learning. The course aims to equip students with:

- Foundational knowledge and skills: Students will begin with exploring the evolution, fundamental principles, and architecture of HPC systems, understanding resource management, performance metrics, and the operational dynamics of parallel systems. Through hands-on sessions, students will engage with cloud-based HPC applications, emphasizing health and neurosciences, while learning to set up environments and write simple parallel codes.
- Advanced parallel computing techniques: Exploring deeper into parallel computing, students will master distributed and shared memory concepts, MPI programming, and OpenMP for multithreading. A special focus on GPU computing and OpenAcc/CUDA basics will equip students with the skills to handle computational fluid dynamics and integrate MPI with OpenMP in complex simulations.
- Performance optimization: Students will learn to optimize code for various HPC architectures, utilize profiling and performance analysis tools, and understand the importance of memory hierarchy and data locality. The module also covers high-performance libraries, parallel I/O,

and data management strategies, with practical exercises in optimizing performance on HPC clusters.

- Parallel algorithms and machine learning: This module introduces scalable parallel algorithms for big data and machine learning, including data structures for high performance and visualization techniques. Students will gain insights into distributed machine learning and strategies for implementing scalable deep learning models, culminating in a project to implement a distributed ML model.
- Advanced parallel programming: Expanding on parallel programming, this module explores patterns in hybrid computing, advanced MPI, and OpenMP techniques, and the use of accelerators like FPGAs. Practical projects on climate and terrestrial systems modeling using hybrid programming approaches will consolidate learning.
- Exploring the frontiers of HPC: The course concludes by looking towards the future of HPC, including neuromorphic and quantum computing, and the challenges and opportunities of post-exascale computing. A review session will prepare students for the final exam and group assignments.

Upon completion of the course, students will possess a comprehensive overview of HPC in terms of both hardware architecture and software, understand HPC architectures and technologies with a special focus on cluster systems, and be familiar with parallel programming paradigms including MPI, OpenMP, and OpenACC. They will be equipped to measure, assess, and analyse the performance of HPC applications, understanding the role of administration, workload, and resource management. Moreover, students will be able to evaluate the suitability of various HPC solutions for solving complex problems and apply emergent technologies in industry and research settings. Knowledge of emerging computing technologies will enable students to anticipate the future direction of supercomputing.

This course is designed to provide a foundational understanding of High Performance Computing, cultivating in-demand skills and knowledge on techniques and technologies underpinning parallelism and HPC, preparing students for future advancements and professional pursuits in the field.

TEACHING METHODOLOGY

IE University teaching method is defined by its collaborative, active, and applied nature. Students actively participate in the whole process to build their knowledge and sharpen their skills. Professor's main role is to lead and guide students to achieve the learning objectives of the course. This is done by engaging in a diverse range of teaching techniques and different types of learning activities such as the following:

Learning Activity	**Weighting**	**Estimated time a student should dedicate to prepare for and participate in**
Lectures	40.0 %	60.0 hours
Discussions	13.3 %	20.0 hours
Exercises in class, Asynchronous sessions, Field Work	20.0 %	30.0 hours
Group work	10.0 %	15.0 hours
Individual studying	16.7 %	25.0 hours

TOTAL	100.0 %	150.0 hours

AI POLICY

Generative artificial intelligence (GenAI) tools may be used in this course for research, ideation, generating an outline, proofreading and grammar check with appropriate acknowledgement. GenAI may not be used for assignments, coding, group submissions and exams. If a student is found to have used AI-generated content inappropriately, it will be considered academic misconduct, and the student might fail the respective assignment or the course.

If you are in doubt as to whether you are using GenAI tools appropriately in this course, I encourage you to discuss your situation with me.

Below, a suggested format to acknowledge the use of generative AI tools. Please note that acknowledging AI will not impact your grade.

"I acknowledge the use of [AI systems link] to [specify how you used generative AI]. The prompts used include [list of prompts]. The output of these prompts was used to [explain how you used the outputs in your work]"

If AI was permitted to use in your assignment, but you have chosen not to include any AI generated content, the following disclosure is recommended:

"No content generated by AI technologies has been used in this assignment."

PROGRAM

FOUNDATIONAL KNOWLEDGE AND SKILLS

Students will begin with exploring the evolution, fundamental principles, and architecture of HPC systems, understanding resource management, performance metrics, and the operational dynamics of parallel systems. Through hands-on sessions, students will engage with cloud-based HPC applications, emphasizing health and neurosciences, while learning to set up environments and write simple parallel codes.

SESSION 1 (LIVE IN-PERSON)

1.1 Evolution and fundamentals of HPC

This opening session will lay the groundwork for the entire course, detailing the structure, assignments, grading, and class dynamics. We will dive into the core concepts of high-performance computing, exploring the evolution of HPC, the fundamental anatomy of supercomputers, key performance indicators, and a brief historical journey through the development of supercomputing. The methodology for tackling both group and individual assignments will be introduced, preparing students on a path to understanding and leveraging HPC technologies effectively.

SESSION 2 (LIVE IN-PERSON)

1.2 Architectural overview of HPC systems

Our exploration continues with a comprehensive look at the architecture of HPC systems. This session will demystify the various parallel architecture families, grounded in Flynn's taxonomy, and discuss the technologies enabling supercomputing today. We will examine diverse architectures and processors, culminating in an understanding of multiprocessors and heterogeneous computing structures. This foundation is crucial for appreciating the complexity and power of modern HPC systems.

SESSION 3 (LIVE IN-PERSON)

1.3 Resource management and performance metrics in parallel systems

In this session, we'll explore the operational aspects of HPC systems, focusing on resource management and the critical performance metrics that guide the efficient use of parallel computing resources. Topics will include job scheduling strategies, resource allocation techniques, and understanding performance benchmarks that ensure the optimal operation of HPC environments. This session aims to equip students with the knowledge to navigate and utilize HPC resources effectively for their computational tasks.

SESSION 4 (LIVE IN-PERSON)

1.4 Introduction to Cloud-based HPC and Virtualization

The advent of cloud computing has revolutionized the accessibility and scalability of HPC resources. This session introduces cloud-based HPC solutions, covering the basics of virtualization, cloud service models, and the benefits and challenges of cloud HPC. We will explore how virtualization technologies underpin the flexibility of cloud resources and discuss strategies for integrating cloud-based HPC into scientific workflows, providing a modern perspective on HPC utilization.

SESSION 5 (ASYNCHRONOUS)

1.5 HPC in Health & Neurosciences. Environment Setup and Simple Parallel Codes

Focusing on the application of HPC in critical research areas, this asynchronous session will guide students through setting up HPC environments and writing simple parallel codes, with specific examples from health and neuroscience research. Through online resources and assignments, students will learn to apply HPC concepts to real-world problems, demonstrating the profound impact of computational power in advancing scientific discoveries. This session will combine theoretical knowledge with practical skills, bridging the gap between HPC technology and its applications in groundbreaking research areas.

Individual and Group Assignments Explanation.

PARALLEL COMPUTING ESSENTIALS

Exploring deeper into parallel computing, students will master distributed and shared memory concepts, MPI programming, and OpenMP for multithreading. A special focus on GPU computing and OpenAcc/CUDA basics will equip students with the skills to handle computational fluid dynamics and integrate MPI with OpenMP in complex simulations.

SESSION 6 (LIVE IN-PERSON)

2.1 Distributed and shared memory concepts

This session examines the core principles of parallel computing architectures, focusing on distributed and shared memory systems. Students will learn the differences between these two fundamental architectures, including their respective advantages, challenges, and suitable application scenarios. Through examples and case studies, we'll explore how these memory models influence programming paradigms and performance outcomes, laying the groundwork for understanding more complex parallel computing strategies.

SESSION 7 (LIVE IN-PERSON)

2.2 Deep Dive into MPI programming

Message Passing Interface (MPI) is a cornerstone of scalable parallel computing. In this intensive session, students will be introduced to MPI programming, covering basic to intermediate concepts and commands necessary for developing efficient parallel applications. Through hands-on exercises, participants will gain practical experience in writing and executing MPI programs, focusing on communication patterns, data distribution, and collective operations, which are essential for leveraging distributed-memory systems.

SESSION 8 (LIVE IN-PERSON)

2.3 OpenMP for multithreading

Focusing on shared-memory parallelism, this session introduces OpenMP, a popular API for multithreaded programming. Students will learn how to utilize OpenMP directives to parallelize code efficiently for multicore processors, including workload distribution, synchronization, and memory management techniques. Interactive examples will allow students to experiment with OpenMP to solve computational problems, highlighting strategies for achieving optimal performance on shared-memory architectures.

SESSION 9 (LIVE IN-PERSON)

2.4 Introduction to GPU Computing, OpenAcc/CUDA basics

As computational demands in scientific research and industry grow, GPUs have emerged as powerful accelerators for parallel computing. This session introduces GPU architecture and programming, with a focus on CUDA and OpenACC as tools for harnessing GPU capabilities. Students will explore the basics of GPU computing, learning to develop and optimize parallel algorithms that leverage the massive parallelism of GPUs for computationally intensive tasks.

SESSION 10 (ASYNCHRONOUS)

2.5 Computational Fluid Dynamics (CFD) with HPC. MPI and OpenMP Code Integration and Execution

Applying parallel computing principles to real-world scientific challenges, this session focuses on Computational Fluid Dynamics (CFD), a field that greatly benefits from HPC technologies. Through asynchronous learning materials, students will engage with CFD simulations, employing MPI and OpenMP to tackle complex fluid dynamics problems. This session will provide practical experience in integrating MPI and OpenMP for hybrid parallel programming, demonstrating the power of distributed and shared memory paradigms for high-performance scientific computing.

PERFORMANCE OPTIMIZATION IN HPC

Students will learn to optimize code for various HPC architectures, utilize profiling and performance analysis tools, and understand the importance of memory hierarchy and data locality. The module also covers high-performance libraries, parallel I/O, and data management strategies, with practical exercises in optimizing performance on HPC clusters.

SESSION 11 (LIVE IN-PERSON)

3.1 Optimizing code for HPC architectures

This session digs into the strategies and techniques for optimizing computational codes to fully exploit the capabilities of HPC architectures. We will explore the importance of code profiling, algorithmic choices, and data structures optimization. Key topics include loop unrolling, vectorization, memory access patterns, and cache utilization.

SESSION 12 (LIVE IN-PERSON)

3.2 Profiling and performance analysis tools. Checkpointing

Understanding the performance characteristics of HPC applications is crucial for effective optimization. This session introduces a range of tools and methodologies for performance profiling and analysis, covering both CPU and memory performance aspects. Additionally, we will discuss the concept and techniques of checkpointing as a means to enhance fault tolerance in long-running computations. Through examples, students will learn to identify bottlenecks and apply strategies to mitigate them, ensuring robust and efficient HPC applications.

SESSION 13 (LIVE IN-PERSON)

3.3 Memory hierarchy and data locality.High-Performance libraries for scientific computing

Optimizing for the memory hierarchy and ensuring data locality are key to unlocking the full potential of HPC systems. This session covers the principles of memory hierarchy, from registers to disk storage, and strategies to maximize data locality in computational codes. We will also explore the use of high-performance scientific libraries that provide optimized implementations of common mathematical operations, enabling developers to leverage community knowledge and resources for efficient HPC application development.

SESSION 14 (LIVE IN-PERSON)

3.4 Parallel I/O and high throughput data management. Parallel filesystems

As HPC applications scale, efficient input/output (I/O) and data management become increasingly critical. This session addresses the challenges and solutions associated with parallel I/O operations, including the use of parallel filesystems designed to support high-throughput data access. Students will learn about the architecture of parallel filesystems, best practices for scalable I/O operations, and techniques for managing large datasets in distributed HPC environments.

SESSION 15 (ASYNCHRONOUS)

3.5 Molecular systems & material sciences. Performance tuning on HPC clusters

Focusing on the application of HPC in molecular systems and material sciences, this session offers students the opportunity to apply performance optimization techniques in a real-world scientific context. Asynchronous materials will guide students through case studies involving computational chemistry and physics, highlighting the role of HPC in simulating and understanding complex molecular interactions and material properties. Practical assignments will involve tuning performance of simulation codes on HPC clusters, reinforcing the skills and concepts learned throughout the module.

PARALLEL ALGORITHMS AND MACHINE LEARNING

This module introduces scalable parallel algorithms for big data and machine learning, including data structures for high performance and visualization techniques. Students will gain insights into distributed machine learning and strategies for implementing scalable deep learning models, culminating in a project to implement a distributed ML model.

SESSION 16 (LIVE IN-PERSON)

4.1 Scalable parallel algorithms for Big Data. MapReduce

This session introduces scalable parallel algorithms essential for processing big data efficiently on HPC systems. We will cover the fundamentals of designing algorithms that can scale across multiple processing units, focusing on the MapReduce programming model as a cornerstone for distributed data processing. Through examples and case studies, students will learn how parallel algorithms are applied to real-world big data challenges, enabling insights and discoveries at unprecedented scales.

SESSION 17 (LIVE IN-PERSON)

4.2 Data structures for High Performance. Visualization

Optimized data structures are critical for achieving high performance in computing applications. This session explores the design and implementation of data structures that maximize efficiency in parallel and distributed environments. Additionally, we delve into the principles and tools for scientific visualization, a key aspect of interpreting complex datasets generated by HPC applications. Through practical exercises, students will gain experience in implementing efficient data structures and visualizing data for analysis and presentation.

SESSION 18 (LIVE IN-PERSON)

4.3 Fundamentals of distributed machine learning

With the rise of machine learning, leveraging HPC for distributed training has become increasingly important. This session covers the foundational concepts of distributed machine learning, including data parallelism, model parallelism, and strategies for scaling machine learning algorithms across multiple computing nodes. Students will be introduced to frameworks and libraries that facilitate distributed machine learning, preparing them for the development of scalable AI models.

SESSION 19 (LIVE IN-PERSON)

4.4 Deep learning at scale

Building on the previous session, we focus on scaling deep learning models using HPC resources. This session examines the challenges and solutions associated with training large neural networks, highlighting techniques such as gradient accumulation, batch size scaling, and the use of accelerators like GPUs and TPUs. Through examples, students will learn to implement and train deep learning models efficiently on HPC systems, exploring the latest advancements in scalable AI technologies.

SESSION 20 (ASYNCHRONOUS)

4.5 Implementing a Distributed ML Model

In this asynchronous session, students will apply the concepts learned about distributed machine learning and deep learning at scale to implement a distributed machine learning model. This session emphasizes the practical application of distributed ML techniques, fostering a deep understanding of HPC's role in advancing machine learning research and applications.

ADVANCED PARALLEL PROGRAMMING

Expanding on parallel programming, this module explores patterns in hybrid computing, advanced MPI, and OpenMP techniques, and the use of accelerators like FPGAs. Practical projects on climate and terrestrial systems modeling using hybrid programming approaches will consolidate learning.

SESSION 21 (LIVE IN-PERSON)

5.1 Patterns in hybrid computing

This session explores the intricate world of hybrid computing, where different parallel computing paradigms coexist and complement each other within the same application. Students will learn about the common patterns used in hybrid computing, including the integration of MPI with OpenMP and GPU computing. Through case studies and examples, the session will highlight how these patterns can be leveraged to solve complex computational problems more efficiently by combining the strengths of each computing paradigm.

SESSION 22 (LIVE IN-PERSON)

5.2 Advanced MPI

Building upon the foundational MPI concepts covered in earlier modules, this session delves into advanced MPI features and techniques. Topics will include dynamic process management, one-sided communications, and the use of persistent communication requests. These advanced topics are essential for optimizing communication in large-scale parallel applications, and students will have the opportunity to apply these concepts through guided programming exercises.

SESSION 23 (LIVE IN-PERSON)

5.3 Advanced OpenMP

Continuing the exploration of parallel programming, this session focuses on advanced OpenMP features, including task-based parallelism, SIMD directives, and the management of device memory in the context of GPU accelerators. Students will learn to enhance the performance and scalability of shared-memory parallel applications by employing these advanced OpenMP capabilities, preparing them to tackle more complex computational challenges.

SESSION 24 (LIVE IN-PERSON)

5.4 Accelerators: FPGAs and specialized processors

This session introduces Field-Programmable Gate Arrays (FPGAs) and other specialized processors as accelerators in high-performance computing. Students will learn about the architecture of FPGAs, their programming models, and how they can be used to accelerate specific computational tasks. The session will also cover the role of other specialized processors in HPC, providing insights into selecting the right accelerator for different types of computational workloads.

SESSION 25 (ASYNCHRONOUS)

5.5 Climate and terrestrial systems modelling. Hybrid MPI/OpenMP programming

In this session, students will apply their knowledge of advanced parallel programming and accelerators to tackle a real-world problem in the domain of climate and terrestrial systems modeling. Using asynchronous materials and assignments, students will work on projects that require the integration of MPI and OpenMP, possibly employing GPU accelerators or FPGAs to enhance performance. This practical application reinforces the skills learned throughout the module and demonstrates the impact of advanced parallel programming techniques on solving critical environmental challenges.

FRONTIERS OF HPC

Looking towards the future of HPC, including neuromorphic and quantum computing, and the challenges and opportunities of post-exascale computing. A review session will prepare students for the final exam and group assignments.

SESSION 26 (LIVE IN-PERSON)

6.1 Neuromorphic computing

This session introduces the innovative field of neuromorphic computing, inspired by the structure and functionality of the human brain. Students will learn about the design and application of neuromorphic systems, which mimic neural architectures to achieve high efficiency in specific computational tasks, such as sensory data processing and machine learning. Through discussions and examples, the potential impact and current limitations of neuromorphic computing in scientific research and technology development will be explored.

SESSION 27 (LIVE IN-PERSON)

6.2 Quantum computing fundamentals and applications

Quantum computing represents a profound shift in computational paradigms, offering the potential to solve problems that are intractable for classical computers. This session covers the basics of quantum computing, including qubits, quantum gates, and entanglement. Students will also be introduced to quantum algorithms and their applications in cryptography, optimization, and simulation. The session aims to demystify quantum computing and discuss its implications for the future of HPC and beyond.

SESSION 28 (LIVE IN-PERSON)

6.3 Advanced computing: Toward the post-exascale era

As HPC systems approach and surpass the exascale performance milestone, this session looks forward to the next generation of challenges and opportunities in advanced computing. Topics include emerging architectures, software paradigms, and the integration of AI and big data analytics into HPC workflows. Students will engage in discussions on the technological and scientific considerations as computing moves beyond exascale, preparing them for the evolving landscape of high-performance computing.

SESSION 29 (LIVE IN-PERSON)

6.4 Review Session Pre-exam

This session is dedicated to reviewing the key concepts, technologies, and applications covered throughout the course. It will provide students with the opportunity to clarify doubts, discuss challenging topics, and synthesize the knowledge acquired in previous modules. The review session will prepare students for the final examination, ensuring a comprehensive understanding of the course material and readiness for practical application and further exploration in the field of HPC.

SESSION 30 (LIVE IN-PERSON)

Exam and Group Assignments

The final session of the course is dedicated to the examination and presentation of group assignments. Students will demonstrate their mastery of high-performance computing concepts, techniques, and applications through a written exam and the presentation of group projects. These projects will encompass the design, implementation, and optimization of HPC solutions for real-world problems, showcasing the practical skills and innovative approaches developed by students throughout the course.

EVALUATION CRITERIA

The evaluation of this course is designed to assess both theoretical understanding and practical application of concepts related to High-Performance Computing (HPC). The breakdown ensures a balanced approach through continuous assessment, individual accountability, teamwork, and examinations.

Class participation: Active participation in discussions and activities is essential for understanding real-world HPC applications. Attendance is mandatory and contributes to the overall grade.
Individual work: Each module includes an individual assignment. These assignments focus on applying concepts from the lectures to real-world problems or case studies.
Intermediate tests: Short quizzes will be administered regularly to evaluate comprehension of recent lectures. These tests reinforce progressive learning.
Group work: Students will complete a group project including a written paper and an oral presentation on a selected HPC-related topic.
Mid-course exam: A written individual exam to be held mid-semester covering content from the first half of the course.
Final exam: A comprehensive exam at the end of the course evaluating both theoretical and applied understanding of all modules.

criteria	**percentage**	**Learning Objectives**	**Comments**
Class Participation	10 %		
Individual Work	15 %		
Intermediate Tests	15 %		Quizzes and intermediate exam
Workgroups	15 %		
Final Exam	30 %		
Mid-course exam	15 %		

RE-SIT / RE-TAKE POLICY

Each student has four chances to pass any given course distributed over two consecutive academic years: ordinary call exams and extraordinary call exams (re-sits) in June/July.

Students who do not comply with the 80% attendance rule during the semester will fail both calls for this Academic Year (ordinary and extraordinary) and have to re-take the course (i.e., re-enroll) in the next Academic Year.

Evaluation criteria:

- Students failing the course in the ordinary call (during the semester) will have to re-sit the exam in June / July (except those not complying with the attendance rule, who will not have that opportunity and must directly re-enroll in the course on the next Academic Year).
- The extraordinary call exams in June / July (re-sits) require your physical presence at the campus you are enrolled in (Segovia or Madrid). There is no possibility to change the date, location or format of any exam, under any circumstances. Dates and location of the June / July re-sit exams will be posted in advance. Please take this into consideration when planning your summer.
- The June / July re-sit exam will consist of a comprehensive exam. Your final grade for the course will depend on the performance in this exam only; continuous evaluation over the semester will not be taken into consideration. Students will have to achieve the minimum passing grade of 5 and can obtain a maximum grade of 8.0 (out of 10.0) – i.e., “notable” in the in the re-sit exam.
- Retakers: Students who failed the subject on a previous Academic Year and are now re-enrolled as re-takers in a course will be needed to check the syllabus of the assigned

professor, as well as contact the professor individually, regarding the specific evaluation criteria for them as retakers in the course during that semester (ordinary call of that Academic Year). The maximum grade that may be obtained in the retake exam (3rd call) is 10.0.

After ordinary and extraordinary call exams are graded by the professor, you will have a possibility to attend a review session for that exam and course grade. Please be available to attend the session in order to clarify any concerns you might have regarding your exam. Your professor will inform you about the time and place of the review session. Any grade appeals require that the student attended the review session prior to appealing.

- Students failing more than 18 ECTS credits in the academic year after the June-July re-sits will be asked to leave the Program. Please, make sure to prepare yourself well for the exams in order to pass your failed subjects.
- In case you decide to skip the opportunity to re-sit for an exam during the June / July extraordinary call, you will need to enroll in that course again for the next Academic Year as a re-taker and pay the corresponding extra cost. As you know, students have a total of four allowed calls to pass a given subject or course, in order to remain in the program.

BIBLIOGRAPHY

Compulsory

- Sterling, T., Anderson, M., Brodowicz, M., & Bell, C. G. (2018). *High performance computing: Modern systems and practices.* Cambridge, MA : Morgan Kaufmann. ISBN 9780124201583 (Digital)

High Performance Computing: Modern Systems and Practices is a fully comprehensive and easily accessible treatment of high performance computing, covering fundamental concepts and essential knowledge while also providing key skills training. With this book, domain scientists will learn how to use supercomputers as a key tool in their quest for new knowledge.

Recommended

- Robert Robey and Yuliana Zamora. (2021). *Parallel and High Performance Computing.* Manning Publications. ISBN 9781617296468 (Digital)

Parallel and High Performance Computing offers techniques guaranteed to boost your code’s effectiveness. You’ll learn to evaluate hardware architectures and work with industry standard tools such as OpenMP and MPI. You’ll master the data structures and algorithms best suited for high performance computing and learn techniques that save energy on handheld devices. You’ll even run a massive tsunami simulation across a bank of GPUs.

- Taylor & Francis Group. (2019). *Introduction to High Performance Computing for Scientists and Engineers.* Taylor & Francis Group. ISBN 9780367221300 (Digital)

Written by high performance computing (HPC) experts, Introduction to High Performance Computing for Scientists and Engineers provides a solid introduction to current mainstream computer architecture, dominant parallel programming models, and useful optimization strategies for scientific HPC. From working in a scientific computing center, the authors gained a unique perspective on the requirements and attitudes of users as well as manufacturers of parallel computers.

- Barba, L. A., & Forsyth, G.. (2021). *High Performance Python: Practical Performant Programming for Humans..* O'Reilly Media. ISBN 9781492055020 (Digital)

This book focuses on optimizing Python code for performance, with particular emphasis on numerical algorithms and data-intensive applications. It's a valuable resource for those looking to use Python in HPC contexts.

BEHAVIOR RULES

Please, check the University's Code of Conduct here. The Program Director may provide further indications.

ATTENDANCE POLICY

Please, check the University's Attendance Policy here. The Program Director may provide further indications.

ETHICAL POLICY

Please, check the University's Ethics Code here. The Program Director may provide further indications.