UAB
Universitat Autònoma de Barcelona

High-performance Computation

2026/2027

Code: 104404
Credits: 6

Degree programme	Type	Course
Computational Mathematics and Data Analytics	OB	3

Contact lecturer

Name : Carles Carrillo Jordan

Email : carles.carrillo@uab.cat

Teaching staff

Carles Carrillo Jordan

Christian Guzman Ruiz

Group languages

You can consult this information at the end of the document.

Prerequisites

Although there are no formally established prerequisites and the subject will provide students with the means to acquire the knowledge described in the syllabus, it is advisable: a good knowledge of programming, of the computer structure, of the operating system at the level of user programmer and how messages are sent over the network (Fondations of Computers, System Software, Introduction to Programming)

Objectives

The objective of this subject is to study high performance and parallel computing systems, multiprocessor and multicomputer systems, paradigms of parallel programming, learn to develop applications with shared memory, introduce the concept of message passing and applications in distributed memory, and analyze the performance of these applications.

The theoretical concepts about programming paradigms, shared memory and message passing, are reinforced with problem sessions and labs in which students learn to program using parallel programming languages.

All the components described in this subject should allow the student to understand the performance of high performance and parallel systems and, to a certain extent, be able to perform a simple design of a parallel application and evaluate its performance.

Learning outcomes

- CM28 (Design effective algorithmic solutions to computational problems in accordance with the established requirements.) Design effective algorithmic solutions to computational problems in accordance with the established requirements.

- CM29 (Assess the computational complexity of the algorithmic solutions to develop and implement the one that guarantees the best performance.) Assess the computational complexity of the algorithmic solutions to develop and implement the one that guarantees the best performance.
- KM23 (Identify the suitable programming strategies to solve a given problem.) Identify the suitable programming strategies to solve a given problem.
- SM25 (Apply the characteristics, functionalities and structure of hardware Distributed Systems and Computer Networks to design and implement applications based on them.) Apply the characteristics, functionalities and structure of hardware Distributed Systems and Computer Networks to design and implement applications based on them.
- SM26 (Apply the fundamental principles and basic techniques of parallel, concurrent and distributed programming.) Apply the fundamental principles and basic techniques of parallel, concurrent and distributed programming.
- SM27 (Assess computer architectures, including parallel and distributed platforms, developing and optimising software for these.) Assess computer architectures, including parallel and distributed platforms, developing and optimising software for these.

Contents

Topic 1: C Programming Review

C Review and Integration of C Code with Python.

Topic 2: Basic Concepts

SIMD (Single Instruction, Multiple Data), MIMD (Multiple Instruction, Multiple Data). Parallel Application Models.

Concurrency: Concept, Race Conditions, Critical Region and Mutual Exclusion Mechanisms.

Topic 3: Design and Evaluation of Parallel Applications

Introduction to High Performance Systems, Parallel Systems, Multiprocessors and Multicomputers.

Evaluating the Performance of Parallel Implementations.

Topic 4: OpenMP

Shared Memory Applications. OpenMP (Open Multiprocessing) Standard.

Topic 5: OpenACC - Cuda

Development of Applications Using OpenACC and CUDA.

Topic 6: MPI

Message Passing Applications. MPI (Message Passing Interface) standard.

Learning activities and methodology

Title	Hours	ECTS	Learning outcomes
Seminar preparation	30	1.2	CM28, CM29, KM23, SM25, SM26, SM27
Autonomous preparation	30	1.2	CM28, CM29, KM23, SM25, SM26, SM27
Lab preparation	30	1.2	CM28, CM29, KM23, SM25, SM26, SM27
Exercise Seminars	11	0.44	CM28, CM29, KM23, SM25, SM26, SM27
Labs	10	0.4	CM28, CM29, KM23, SM25, SM26, SM27
Theory	24	0.96	CM28, CM29, KM23, SM25, SM26, SM27

During the development of the subject four types of teaching activities cam be differentiated:

- Theoretical classes: explanation on the blackboard of the theoretical part of each program topic. The typical structure of a theoretical lesson will be the following: first, an introduction will be made where the objectives of the lesson and the contents to be discussed will be briefly presented. Next, the contents of the lesson will be discussed, including narrative expositions, formal developments that provide the

theoretical foundations, including examples that illustrate the application of the discussed contents. Finally, the professor will present the conclusions of the lesson. Throughout the course there will be continuous assessments of groups of topics.

- Practical classes: All topics will be accompanied by a list of problems that the student should try to solve. In this sense, and as the student progresses and deepens in his/her knowledge, these problems will be more and more complex. The seminars will be the natural forum in which the development of practical work can be discussed, contributing the knowledge that the student lacks in order to carry it forward. The mission of the practical classes is to bridge the theoretical classes and the laboratory classes, which will promote the capacity for analysis and synthesis, critical reasoning, and that will train the student in solving problems. Those that the teacher considers of greater interest or in which the students find more difficulty will be developed on the blackboard. Before the beginning of each topic the professor may propose a list of exercises that the students will have to solve and deliver.
- Laboratory classes: The practical part of the theoretical topics will be completed with lab sessions, where the student will develop a series of programs and should try to solve a specific problem that will be received at the beginningof the semester. Some of these exercises must be delivered on the specified dates. The lab sessions will be developed in groups of two or three students. The subject includes a least 6 sessions in the laboratory, lasting 2 hours each, where the student will carry out the exercises.

This approach to work is oriented to promote an active learning and developing competencies of organization and planning skills, oral and written communication, teamwork and critical reasoning. The quality of the exercises carried out, their presentation and their functioning will be especially valued.

Transversal competences:

As mentioned before, the subject includes as a teaching activity the realization of a writteng assignment that consists in the development of an article to the Vikepèdia which is oriented to communicate efficiently knowledge, results and skills, both in professional environments non expert audiences, making efficient use of ICT in communication and the transmission of ideas.

The teaching management of the subject will be done through the Virtual Campus, which will serve to see the materials, manage the groups of practices, make the corresponding deliveries, see the grades, communicate with professors, etc.

Annotation: within the schedule set by the centre or degree programme, 15 minutes of one class will be reserved for students to evaluate their lecturers and their courses or modules through questionnaires.

Assessment

Continuous assessment activities

Title	Weight	Hours	ECTS	Learning outcomes
Exercises assessment	30%	5	0.2	CM28, CM29, KM23, SM25, SM26, SM27
2nd Individual exam	20%	3	0.12	CM28, CM29, KM23, SM25, SM26, SM27
Laboratories	30%	4	0.16	CM28, CM29, KM23, SM25, SM26, SM27
1st Individual exam	20%	3	0.12	CM28, CM29, KM23, SM25, SM26, SM27

The objective of the assessment process is to verify that the student has achieved the knowledge and skills defined in the objectives of the subject, as well as the associated competences.

Four types of activities will be assessed independently, and the weighted sum of them will give the final grade. These activities are:

1. Theory (T)
2. Solution of the laboratory practices (PL)
3. Completion of practical exercises (PA)

The Theory (T) part will be assessed with two individual partial exams throughout the course. The final grade of Theory will come from the weighted sum of the two exams (0.5 * Control 1 + 0.5 * Control 2). There will be a second chance to recover that part on the day at June we have assigned to. Parts that have not been passed in the partial theory exams may be recovered separately. The minimum grade to pass this part of Theory is > = 5. In the recovery only the failed exams can be recovered, the grade obtained cannot be raised (in the case of passing the Theory part). The maximum grade that can be obtained in recovery is 7.

The part of Laboratory exercises (PL) will be evaluated by group. There are three deliveries. The final grade will come from the weighted sum of the three deliveries (X0 * Delivery 1 + X1 * Delivery 2 + X2 * Delivery 3, X0 + X1 + X2 = 1). To pass the PL the minimum mark will have to be 5. There is only one opportunity to pass this part (this part can not be recovered). There will be 3 short and written exams for individual validation of the practical projects in the class schedule (the third exam in the schedule established by the coordination for the second partial exams in June). Practical projects (PL) = Lab * Validation. To pass the PL the minimum mark will have to be >= 5. These sessions are mandatory.

The practical exercises (PA) will consist of solving very specific programming problems that will be relate in some cases to those that will be found in the laboratory exercices. It is expected that each student will focus in the resolution of a set of specific problems isolated from the most general case of the lab exercices. The value of these exercises is 30% of the final mark and given their nature and objective are not recoverable.

The final grade of the subject will be the weighted sum of the grades of each of the four activities: 40% of Theory, 30% Resolution of individual practical exercises and 30% of Resolution of laboratory exercices. The result will to pass the subject have to be >= 5.

In case a student does not pass the subject due to not reaching the minimum score in any of the mandatory parts (Theory or Laboratory exercices), even though the weighted average is equal or superior to 5, the final grade of the subject will be 4.5.

In the event that the average does not reach 5, the official grade will be the average mark obtained numerically.

If the student delivers any activity, it is understood that he/she is participating in the subject and will be evaluated. If you do not deliver any activity, then it can be considered Non-evaluable.

Granting an honorific matriculation qualification is a decision of the faculty responsible for the subject. The regulations of the UAB indicate that MH can only be awarded to students who have obtained a final grade of 9.00 or more. It can be granted up to 5% of MH of the total number of students enrolled.

The dates of continuous evaluation and assignment delivery will be published on the virtual campus and may be subject to possible changes to adapt to possible incidents; the virtual campus will always inform about these changes since it is understood that the CV is the usual mechanism for exchanging information betweenprofessors and students.

For each assessment activity, a place, date and time of revision will be indicated in which the student will be able to review the activity with the professor. In this context, claims can be made about the activity grade,which will be evaluated by the professors responsible for the subject. If the student does not submit to this review, this activity will not be reviewed later.

SUMMARY

If ((T> = 5) i (P> = 5) then

NF = 0.4 * T + 0.3 * PA + 0.3 * PL

If (NF> = 5) then PASS

elseFAIL

Repeating students: repeating students who have passed laboratory exercices in previous editions of the subject may request the validation of this part of the subject. The rest of the assessment activities must be carried out under the same conditions as the other students.

Single Assessment Evaluation: This subject DOES NOT HAVE SINGLE ASSESSMENT EVALUATION.

Note about plagiarism:

Without prejudice to other disciplinary measures deemed appropriate, and in accordance with the current academic regulations, irregularities committed by a student who maylead to a variation of the qualification in an assessable activity will be graded with zero (0). Assessment activities qualified in this way and by this procedure will not be recoverable. If it is necessary to pass any of these assessment activities to pass the subject, this subject will be suspended directly, without opportunity to recover it in the same course. These irregularities include, among others:

- the total or partial copy of a lab exercice, report, or any other evaluation activity;
- let another student to copy;
- present a group work not done entirely by the members of the group (applied to all members and not only to those who have not worked);
- present as own materials prepared by a third party, even if they are translations oradaptations, and generally works with non-original and exclusive elements of the student;
- have communication devices (such as mobilephones, smart watches, pens with camera, etc.) accessible during theoretical-practical assessment tests (individual exams);
- talk with classmates during the individual theoretical-practice tests (exams);
- copy or attempt to copy from other students during the theoretical-practical assessment tests (exams);
- use or attempt to use written material related to the subject during the theoretical-practical evaluation tests (exams), when these have not been explicitly allowed.

If you do not pass the subject due to the fact that some of the evaluation activities do not reach the minimum grade required, the numerical official grade will be the lowest value between 4.5 and the weighted average of the grades. With the exceptions that the \"Non-Appraising\" qualification will be awarded to students who do not participate in any of the assessment activities, and that the numerical official grade will be the lowest value between 3.0 and the average Weighted grades in case the student has committed irregularities in an evaluation act (and therefore the subject cannot beapproved by compensation). In future editions of this subject, the student who has committed irregularities in an evaluation act will not be validated any of the assessment activities carried out.

In summary: copy, let copy or plagiarize (or attempt) in any of the assessment activities will lead to a FAIL, not compensable and without validations of parts of the subject in subsequent courses.

In this subject, the use of Artificial Intelligence (AI) technologies is allowed exclusively in support tasks, such as bibliographic or information searches, text correction or translations. The student must clearly identify which parts have been generated with this technology, specify the tools used and include a critical reflection on how these have influenced the process and the final result of the activity. The lack of transparency in the use of AI in this assessable activity will be considered a lack of academic honesty and may lead to a partial or total penalty in the grade of the activity, or greater sanctions in serious cases.

Bibliography

- Parallel Programming Concepts and Practice. Bertil Schmidt, Jorge González-Domínguez, Christian Hundt, Moritz Schlarb. Morgan Kaufmann, 2018
- An Introduction to Parallel Programming, 2nd edition. Peter S. Pacheco. Morgan Kaufmann. 2018
- Programming Massively Parallel Processors, A Hands-on Approach. David B. Kirk, Wen-mei W. Hwu. Morgan Kaufmann. 3th Edition. 2018

- Computer Architecture. A quantitative approach. John L. Hennessy, David A. Patterson. Morgan Kaufmann. 6a edición. 2018
- Introduction to Parallel Computing. A. Grama et alter. Addison Wesley, Second Edition, 2003.
- Parallel Program Development For Cluster Computing: Methodology, Tools and Integrated Environments. Edited by J. C. Cunha, P. Kacsuk, S. C. Winter. Nova Science Publishers, Inc., 2001.
- Parallel Programming with MPI, Peter Pacheco, Morgan Kauffman, 1996
- OpenACC Programming and Best Practices Guide (https://www.openacc.org/sites/default/files/inline-files/OpenACC_Programming_Guide_0.pdf)

Software

In this course you will work on clusters of the Department of Computer Architecture and Operating Systems, consequently, you will need an ssh client to make the connections (integrated in any computer with Linux operating system) and it is highly recommended to use a development environment that allows remote connection (such as MobaXTerm for Windows [https://mobaxterm.mobatek.net/] or Visual Studio Code for all platforms [https://code.visualstudio.com/]).

Course groups and languages

The information provided is provisional until November 30. After this date, you will be able to consult the language of each group through this link. To access the information, you will need to enter the course CODE

Type of teaching	Group	Language	Semester	Shift
(TE) Theory	1	Catalan	first semester	morning-mixed
(PLAB) Practical laboratories	1	Catalan	first semester	morning-mixed
(PLAB) Practical laboratories	2	Spanish	first semester	morning-mixed