Universitat de Lleida

DEGREE CURRICULUM

HIGH PERFORMANCE COMPUTING

Coordination: ONRUBIA PALACIOS, JORDI RICARD

Academic year 2023-24

HIGH PERFORMANCE COMPUTING 2023-24

Subject's general information

Subject name	HIGH PERFORMANCE COMPUTING
Code	103084
Semester	2nd Q(SEMESTER) CONTINUED EVALUATION
Typology	Degree: Master's Degree in Informatics Engineering; Course: 1; Character: COMPULSORY; Modality: Attendance-based
Course number of credits (ECTS)	4.5
Type of activity, credits, and groups	see table below
Coordination	ONRUBIA PALACIOS, JORDI RICARD
Department	COMPUTER ENGINEERING AND DIGITAL DESIGN
Teaching load distribution between lectures and independent student work	30% of the time is leaded by the teacher and 70% is based on autonomous work. The hours of work leaded by the teacher will be face-to-face.
Important information on data processing	Consult [this link] for more information.
Language	English

Typology:

Degree	Course	Character	Modality
Master's Degree in Informatics Engineering	1	COMPULSORY	Attendance-based

Type of activity, credits, and groups:

Activity type	PRALAB	TEORIA
Number of credits	3	1.5
Number of groups	1	1

Teaching staff	E-mail addresses	Credits taught by teacher	Office and hour of attention
GINE DE SOLA, FRANCESC	francesc.gine@udl.cat	0	
ONRUBIA PALACIOS, JORDI RICARD	jordi.onrubia@udl.cat	4,5	

Subject's extra information

To follow this course, the student should have solid knowledge of structured programming in C language and Architecture and Computer Technology and Operating Systems.

Learning objectives

- Analyse the performance of an HPC computer.
- Know and understand parallel programming paradigms.
- Know the operation and characteristics of HPC architectures.
- Implement and debug parallel applications using the OpenMP, MPI and CUDA programming models.
- Solve numerical algorithms using hybrid OpenMP-MPI and OpenMP-CUDA technologies and evaluate their performance.
- Communicate ideas and concepts in English in an understandable way, writing and speaking.

Competences

University of Lleida strategic competences

- UdL1: Appropriate skills in oral and written language.
- UdL2: Command of a foreign language.

Cross-disciplinary Competences EPS

- EPS3: Capacity to convey information, ideas, problems and solutions to both a specialized and no specialized public.
- EPS4: Capacity to conceive, design and implement projects and/or contribute to new solutions, using engineering tools.

General Competences

- CG4: Capacity for mathematical modeling, calculation and simulation in technology and engineering company centers, particularly in research, development and innovation tasks in all areas related to informatic engineering.

- CG8: Capacity to apply the knowledge acquired for solving problems in new and unfamiliar situations within broader and more multidisciplinary contexts, and to be capable of integrating this knowledge.

Basic Competences

- CB1: Possess knowledge and understanding that provide a basis or opportunity for originality in developing and / or applying ideas, often in a research context.
- CB2: That the students can apply their knowledge and their ability to solve problems in new or unfamiliar environments within broader (or multidisciplinary) contexts related to their field of study

Degree-specific competences

- CE10: Capacity to understand and apply advanced knowledge in high-performance computing and numerical or computational methods to problems of engineering.

- CE12: Capacity to apply mathematical, statistical and artificial intelligence methods, design and develop applications, services, intelligent systems and systems based on knowledge.

Subject contents

1. Introduction: High Performance Computing (1F + 2h NF)

2. Parallel processing and benchmarking (5h F+13h NF)

- Introduction
- HPC Architectures
- Design of parallel applications
- Performance metrics
- Benchmarking tools

3. Parallel programming in shared memory environments - OpenMP (8h F+21h NF)

- Introduction to OpenMP
- Parallel regions and series
- Environment variables
- Loop decomposition strategies
- Task synchronization
- Memory models

4. Parallel programming in distributed memory environments - MPI (8h F+ 21h NF)

- Introduction to MPI
- Basic structure of MPI programs
- Types of communications
- Derived data types
- Communicators and topologies

5. Parallel programming in GPGPU environments - CUDA (8h F+ 21h NF)

- Introduction to CUDA
- CUDA runtime environment
- Memory management
- Threads, blocks and indexing
- Communications and synchronization
- Device management

- F: Face to Face classes
- NF: Non Face-to-Face

Methodology

Every week, each student will receive:

- **Two hours of Face-to-face lectures** will be done in the lab explaining both, the theorical and practical contents, accompanied by illustrative examples and problem solving. As a support material of the class, we will follow the slides for the course.
- Support materials to follow the subject in a non-attendance way.

The evaluation will be continuous and consist of five different tests:

- 4 practices: Benchmarking, OpenMP, MPI-OpenMP and CUDA.
- Verification through the oral presentation of the hybrid practice (OpenMP-MPI) and CUDA.

Development plan

This scheduling is for guidance only and is subject to possible changes depending on the dynamics of the course.

Week	Description	Activities Leaded by Teacher	Autonomous Activities	Autonomous Hours
1	1 Introduction: High Performance Computing	Presentation of the subject	Review of explained contents	1
	Parallel processing and benchmaking	Lecture and participatory classes	Exercises resolution	1
2	Parallel processing and benchmaking	Master class and laboratory Presentation of SGR queue manager	Exercises resolution Knowledge of SGE queue manager tools	6
3	Parallel processing and benchmaking	Master class and laboratory	Exercises resolution Benchmarking practice	6
4	Parallel programming in shared memory environments - OpenMP	Master class and laboratory	OpenMP Exercises resolution Benchmarking practice	5
5	Parallel programming in shared memory environments - OpenMP	Master class and laboratory	OpenMP Exercises resolution HPC-OpenMP practice Submission benchmarking practice	5
6	Parallel programming in shared memory environments - OpenMP	Master class and laboratory	OpenMP Exercises resolution HPC-OpenMP practice	5
7	Parallel programming in shared memory environments - OpenMP	Master class and laboratory	OpenMP Exercises resolution HPC-OpenMP practice	6
8	Parallel programming in distributed memory environments - MPI	Master class and laboratory	MPI Exercises resolution HPC-OpenMP practice	5

9	Parallel programming in distributed memory environments - MPI	Master class and laboratory	MPI Exercises resolution HPC-OpenMP-MPI practice (Hybrid) Submission of HPC-OpenMP	5
10	Parallel programming in distributed memory environments - MPI	Master class and laboratory	MPI Exercises resolution HPC-OpenMP-MPI practice (Hybrid)	5
11	Parallel programming in distributed memory environments - MPI	Master class and laboratory	MPI Exercises resolution HPC-OpenMP-MPI practice (Hybrid)	6
12	Parallel programming in GPGPU environments - CUDA	Master class and laboratory	CUDA Exercises resolution HPC-CUDA practice	5
13	Parallel programming in GPGPU environments - CUDA	Master class and laboratory	CUDA Exercises resolution HPC-CUDA practice Submission of HPC-OpenMP-MPI (Hybrid)	5
14	Parallel programming in GPGPU environments - CUDA	Master class and laboratory	CUDA Exercises resolution HPC-CUDA practice	5
15	Parallel programming in GPGPU environments - CUDA	Master class and laboratory	CUDA Exercises resolution HPC-CUDA practice	6
16	Assessment period	Tutorial class	HPC-CUDA practice	2
17	Assessment period	Tutorial class	Submission of HPC-CUDA practice	2
18	Assessment period+practice's verification	Tutorial class	Verification of Hybrid and CUDA practices.	3
19	Practice's verification	Oral presentation	Verification of Hybrid and CUDA practices.	3

Evaluation

The evaluation is continuous and is made up of five assessment blocks with the following weights with respect to the final mark for the subject:

- Benchmarking block: 10%
- Shared Memory Computing block: 25%
- Distributed Memory Computing block: 35%
- GPGPU block: 30%

Each block is made up of the following assessment activities.

Assessment block	C/V (1)	Assessment Activity	Weight	I/G (2)

Benchmarking block	C	Benchmarking practice	10%	G
Shared Memory Computing block	C	OpenMP practice	25%	G
Distributed Memory Computing block	C	MPI-Hybrid practice Verification of the practica	35%	G
GPGPU block	C	HPC-CUDA practice Verification of the practice	30%	G

(1) C/V: Compulsory / Voluntary

(2) I/G: Individual / Group

The final mark is calculated as follows:

Final Mark= 10% Benchmarking + 25% OpenMP + 35% OpenMP-MPI (Hybrid practice)*verification +30% CUDA*verification

The verification activity consist on the oral presentation in front of the teacher, carried out during the final semester, where each group should explain the main design decissions taken during the development of the hybrid and CUDA practices, as well a demostration of its righ execution. This presentation will be done in the practice's laboratoy. The mark of this activity will be binary: 1 (Pass) or 0 (Not pass).

Students who fail the continuous assessment with a score lower than 5 shall be entitled to recover the tests with a weight higher than 20%.

The student who can do the alternative assessment (see UdL assessment regulations) must to do the following activities:

The student who does not pass the alternative assessment with a grade equal to or greater than 5 will have the right to recover the activities with a weight greater than 20%.

The student who can do the alternative assessment (see UdL assessment regulations) must to do the following activities:

- HPC-OpenMP practice: 25%
- HPC-MPI-OpenMP practice (Hybrid) and its subsequent verification carried out in the same conditions explained on the continuous assessment section: 40%
- HPC-CUDA practice its subsequent verification carried out in the same conditions explained on the continuous assessment section: 35%

Bibliography

F. Giné & J.L. Lérida

Slides of the Subject

Lleida, 2020

P.S. Pacheco,

Parallel Programming with MPI,

Morgan Kaufmann Publishers , 1997

R. Chandra, L. Dagum, D. Kohr,

Parallel Programming in OpenMP,

Morgan Kaufmann Publishers , 2001

S. Duane, Y. Mete

CUDA for Engineers: An Introduction to High-Performance Parallel Computing,

Addison Wesley, 2015

J. Cheng, M. Grossman, T. McKercher

PROFESSIONAL CUDA® C Programming,

John Wiley & Sons, Inc, 2014

https://www.cs.utexas.edu/~rossbach/cs380p/papers/cuda-programming.pdf