Programming

Academic Year: (2023 / 2024) **Review date: 07-06-2023**

Department assigned to the subject: Computer Science and Engineering Department

Coordinating teacher: FERNANDEZ MUÑOZ, JAVIER

Type: Compulsory ECTS Credits : 6.0

Year : 1 Semester : 1

OBJECTIVES

The goal of this course is to introduce to the student in the concept of performance understanding its limits. From this concept, students will learn different techniques to build high performance programs using the C++ programming language as a vehicle.

DESCRIPTION OF CONTENTS: PROGRAMME

1. High Performance Computing Fundamentals.
2. Programming languages and performance
3. Memory management.
4. Generic programming
5. Libraries and interoperability.
6. Code optimization.
7. Applications performance analysis.
8. Concurrent programming
9. Parallel programming

LEARNING ACTIVITIES AND METHODOLOGY

Activities:
* Theoretical Classes: Theoretical presentations accompanied by electronic material such as digital presentations.
* Theoretical Practical classes: Combination of lectures accompanied by the resolution of practical exercises.
* Laboratory practices: Practices to develop in specific laboratories for different subjects.
* Tutoring: Personal student tutoring
* E-learning activities: Course fora, discussion fora and other training e-learning activities.
* Individual student work: individual student activities that complement the other activities and exam preparation.

Teaching methodology
* Teacher class presentations with computer and audiovisual support, in which the main course concepts are developed and bibliography is provided to supplement student learning.
* Resolution of practical cases, problems, etc. posed by the teacher individually or in group
* Expositions and class discussion under the moderation of teacher related to the course content and practical cases.
* Elaboration of reports individually or in group
* Specific e-learning activities, including viewing recorded content, self-correcting activities, participation in fora, and any other mechanism online teaching.

The practical component in this course assingments will be developed for applications performance improvement. Those assignments may include, among others, applications performance analysis, performance problems identification, code optimization, and parallelization techniques.
For these more practical content, attendance to laboratories and individual or group work outside the classroom may be combined with tracking and tutoring students through fora and other mechanisms for discussion. Other e-learning strategies will be also used, such as self-assessment of the work done, all supported through Aula Global.

ASSESSMENT SYSTEM

SE1: In class participation: 10%
SE2-3: Individual work performed during the course: 30%
SE4: Face to face exams: 60%

% end-of-term-examination:	60
% of continuous assessment (assigments, laboratory, practicals…):	40

BASIC BIBLIOGRAPHY

- Anthony Williams C++ Concurrency in Action. Practical Multithreading. 2nd Edition, Manning, 2019
- Bjarne Stroustrup Programming - Principles and Practice. 2nd Edition, Addison-Wesley, 2014
- Michael Voss, Rafael Asenjo, James Reinders Pro TBB: C++ Parallel Programming with Threading Building Blocks, APress, 2019

ADDITIONAL BIBLIOGRAPHY

- Bjarne Stroustrup The C++ Programming Language. 4th Edition, Addison-Wesley, 2013
- Bjarne Stroustrup A Tour of C++. 2nd Edition., Addison-Wesley, 2018
- Gerassimos Barlas Multicore and GPU Programming: An Integrated Approach, Morgan-Kaufmann, 2014
- Kurt Guntheroth Optimized C++, O'Reilly, 2016
- Peter Gottschling Discovering Modern C++: An intensive course for scientists, engineers and programmers, Addison-Wesley, 2015

BASIC ELECTRONIC RESOURCES

- CppReference . C++ Reference: https://en.cppreference.com/w/