Universitat Autònoma de Barcelona

2022/2023

Parallel Programming

Code: 43481
ECTS Credits: 6

Degree	Type	Year	Semester
4313136 Modelling for Science and Engineering	OB	0	1

Contact

Name: Tomás Manuel Margalef Burrull

Email: tomas.margalef@uab.cat

Use of Languages

Principal working language: english (eng)

Teachers

Juan Carlos Moure Lopez

Anna Bàrbara Sikora

Prerequisites

There are no prerequisites

Objectives and Contextualisation

The objective of the module is to:

- Develop C programs
- Identify difficulties related to parallel programming
- Apply an adequate methodology for the development of parallel applications
- Understand the differences of parallel programming approaches: Shared memory, message passing
- Determine the most convinient parallel programming approach to develop a particular application
- Develop parallel applications
- Evaluate parallel application performance and collect necessary measurements to tune the application in order to improve its performance

At the end of this module, students should have enough knowledge, methods and technical skills to develop parallel applications using an adequate programming model and to evaluate the application performance.

Competences

- Analyse and evaluate parallel and distributed computer architectures, and develop and optimise advanced software for these.
- Analyse, synthesise, organise and plan projects in the field of study.
- Apply specific methodologies, techniques and resources to conduct research and produce innovative results in the area of specialisation.
- Continue the learning process, to a large extent autonomously.
- Look for new areas to open up within the field.
- Safeguard, manage, audit and certify the quality of advanced developments, processes, systems and software.

- Take part in research projects and working groups in the field of information engineering and high-performance computation.
- Use acquired knowledge as a basis for originality in the application of ideas, often in a research context.

Learning Outcomes

1. Analyse, synthesise, organise and plan projects in the field of study.
2. Apply specific methodologies, techniques and resources to conduct research and produce innovative results in the area of specialisation.
3. Continue the learning process, to a large extent autonomously.
4. Design the parallel solution to a computational problem taking the characteristics of the hardware available into account.
5. Develop the parallel solution to a computational problem by choosing the most appropriate tools.
6. Identify sources of parallelism in a computational problem.
7. Interpret information from performance-analysis and use this in actions that improve the parallel application.
8. Look for new areas to open up within the field.
9. Plan and develop research projects with content related to parallel programming.
10. Use acquired knowledge as a basis for originality in the application of ideas, often in a research context.
11. Use appropriate tools to analyse the performance of an application.

Content

1. Introduction to the course
2. C programming
3. OpenMP programming
4. MPI programming
5. Performance Analisys
6. GPU Programming

Methodology

The subject will be developed in theoretical classes and practical exercises.

It is recommended that students attend all classes of the subject with a laptop with a well-charged battery.

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.

Activities

Title	Hours	ECTS	Learning Outcomes
Type: Directed			
Attending theoretical lectures and practical exercises	14	0.56	2, 5, 4, 6, 9, 10
Lab practices	24	0.96	5, 4, 7, 9
Type: Autonomous			
Design and development of practical exercises	52	2.08	1, 5, 4, 6, 7, 9, 3, 10, 11
Study	36	1.44	1, 2, 5, 4, 6, 8, 7, 9, 3, 10, 11

Assessment

There are 4 marks corresponding to C programming, OpenMP programming, MPI programming and GPU programming exercises carried out in group. There is also an individual final exam including all the course topics.

Assessment Activities

Title	Weighting	Hours	ECTS	Learning Outcomes
C lab exercise	20	6	0.24	3, 10
Final Exam	30	2	0.08	5, 4, 6, 3, 10
GPU Lab	15	4	0.16	5, 4, 6
MPI Lab exercise	20	8	0.32	1, 2, 5, 4, 6, 8, 7, 9, 3, 10, 11
OpenMP Lab exercise	15	4	0.16	1, 2, 4, 6, 8, 9

Bibliography

Parallel Programming : Techniques and Applications using Networked Workstations and Parallel Computers. Barry Wilkinson. Prentice Hall, 1999.

Designing and Building Parallel Programs: Concepts and Tools for Parallel Software Engineering. Ian Foster . Addison Wesley,1995.

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.

High Performance Cluster Computing (Vols. 1 y 2), Rajkumar Buyya ed., Prentice Hall, 1999.

Parallel Programming with MPI, Peter Pacheco, Morgan Kauffman, 1996

Using MPI-2, William Gropp, Ewing Lusk and Rajeev Thakur, MIT Press, 1999.

Programming Massively Parallel Processors: A Hands-on Approach. David Kirk and Wen-mei Hwu. ISBN: 978-0-12-381472-2. Published by Elsevier Inc. 2010.

http://www.elsevierdirect.com/morgan_kaufmann/kirk/

Software

C

OpenMP

MPI

OpenACC