Parallel and Distributed Calculation Systems 2015/2016

Code: 43343
ECTS Credits: 6

Degree	Type	Year	Semester
4313136 Modelling for Science and Engineering	OT	0	1
4314660 Computer Engineering	OB	1	1

Contact

Name: Elisa Ruth Heymann Pignolo

Email: Elisa.Heymann@uab.cat

Use of languages

Principal working language: english (eng)

Teachers

Miquel Àngel Senar Rosell

Remo Lucio Suppi Boldrito

Antonio Miguel Espinosa Morales

Prerequisites

It is recommended to have successfully completed courses in Computer Basics, Computer Structure, Programming Languages, Operating Systems, Distributed Systems and Networks.

Objectives and Contextualisation

At the end of this subject, students should have enough knowledge, methods and technical skills to research on innovative solutions to distributed systems problems.

Competence	Description
Knowledge	• Analyze and evaluate parallel architectures and distributed computers, and advanced software development and optimization. • Investigate innovative solutions to operating systems problems, servers and applications, and systems based on distributed computing, and more efficient solutions than those currently used. • Understand and analyze the different alternatives for mass storage data systems.
Expertise	• Knowing how to manage parallel computing environments, and understand their implications and cost benefits and services. • Use and apply a wide range of design techniques, middleware and development tools for tuning an application environment. • Be able to select both the distributed platform, such as the most suitable language, for solving problems in distributed computing.

	• Apply the knowledge acquired in the design of distributed storage systems, to design data-intensive applications.
Attitude	• Demonstrate accountability in the management of information and knowledge, and address groups and / or multidisciplinary projects. • Apply research methods, techniques and specific resources for research in a particular area of expertise.

Skills

Modelling for Science and Engineering

- Analyse and evaluate parallel and distributed computer architectures, and develop and optimise advanced software for these.
- Communicate and justify conclusions clearly and unambiguously to both specialised and non-specialised audiences.
- Continue the learning process, to a large extent autonomously.
- Integrate knowledge and use it to make judgements in complex situations, with incomplete information, while keeping in mind social and ethical responsibilities.
- Solve problems in new or little-known situations within broader (or multidisciplinary) contexts related to the field of study.
- 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.

Computer Engineering

- Communicate and justify conclusions clearly and unambiguously to both specialised and non-specialised audiences.
- Communicate orally and in writing in English.
- Continue the learning process, to a large extent autonomously
- Define and communicate results, guaranteeing high levels of performance and quality.
- Design and evaluate operating systems and servers and applications and systems based on distributed computing.
- Display a capacity for the preparation, strategic planning, coordination and technical and financial management of projects in all areas of computer engineering, applying criteria of quality and environmental sustainability.
- Integrate and apply the knowledge acquired and solve problems in new or little-known situations within broader (or multidisciplinary) contexts.
- Integrate knowledge and use it to make judgements in complex situations, with incomplete information, while keeping in mind social and ethical responsibilities.
- Model, design, and define architectures, implement, manage, operate and maintain computer applications, networks, systems, services and content.
- Propose, calculate and design products, processes and installations in all areas of computer engineering.
- Responsibly manage information and knowledge when leading multidisciplinary groups and/or projects.
- Solve problems in new or little-known situations within broader (or multidisciplinary) contexts related to the field of study.
- Use acquired knowledge as a basis for originality in the application of ideas, often in a research context.

Learning outcomes

1. Apply a wide range of techniques for designing middleware and development tools to tie together the environment and the application.
2. Apply the knowledge acquired in the design of distributed storage systems to designing intensive data and computation applications.

3. Apply the knowledge acquired in the design of distributed storage systems to designing intensive data and computing applications.
4. Choose both the distributed platform and the most appropriate language when formulating a solution to a distributed computation problem.
5. Choose both the distributed platform and the most appropriate language when formulating a solution to a distributed computing problem.
6. Communicate and justify conclusions clearly and unambiguously to both specialised and non-specialised audiences.
7. Communicate orally and in writing in English.
8. Continue the learning process, to a large extent autonomously
9. Continue the learning process, to a large extent autonomously.
10. Define and communicate results, guaranteeing high levels of performance and quality.
11. Display a capacity for the preparation, strategic planning, coordination and technical and financial management of projects in all areas of computer engineering, applying criteria of quality and environmental sustainability.
12. Distinguish the parallel computing environments and their implications in terms of performance and cost.
13. Integrate and apply the knowledge acquired and solve problems in new or little-known situations within broader (or multidisciplinary) contexts.
14. Integrate knowledge and use it to make judgements in complex situations, with incomplete information, while keeping in mind social and ethical responsibilities.
15. Propose, calculate and design products, processes and installations in all areas of computer engineering.
16. Responsibly manage information and knowledge when leading multidisciplinary groups and/or projects.
17. Solve problems in new or little-known situations within broader (or multidisciplinary) contexts related to the field of study.
18. Use acquired knowledge as a basis for originality in the application of ideas, often in a research context.

Content

T1: Distributed Computing Platforms (8 hours)

- Data Intensive Applications

- Hadoop framework

- Cluster computing

- Linux Administration (basics: VirtualBox, Hadoop, Slurm)

T2: Scalability, Security, HTC (20 hours: 14h subjects, 2h project discussions, 4h presentation work)

- Scalability: Load balancing, replication, HTC, RMI.

- Security: Assessment, tools.

- Grid Architecture (requirements, key components, resource managers).

T3: Distributed Application Architectures: Cloud computing (CC) (4 hours: 4h subjects)

- Context & short history
- Evolution of Internet computing
- What is Cloud Computing? CC Architecture.
- Characteristics & Types of CC
- Grid Vs. Cloud: Advantages? Downsides?
- Case of study: a) deployment with 2 VM a cloud multisite web service a) deployment with a PaaS of a web service.

T4: Massive Data Handling. Data Bases, from relational DBMS to noSQL recent technologies(8 hours: 4h subjects, 4h lab work)

- noSQL: Big Table, HBase

- MapReduce programming

- Hadoop performance tuning

Methodology

The methodology will combine classroom work, problem solving in class, work in the computing lab, performing works from recommended readings and independent study student. It will use the virtual platform and asked for papers related to the thematic blocks.

Distribution of the tasks:

The Lab work; 50%

Evaluation of work done and presented by the student, presentation work; 50%

Activities

Title	Hours	ECTS	Learning outcomes
Type: Directed			
Lab work	16	0.64	3, 16, 13, 11, 15
Presentation work	8	0.32	7, 10, 16
Subjects	21	0.84	3, 1, 12, 17, 6, 14, 8, 5, 18
Type: Autonomous			
Study and home works	100.5	4.02	3, 1, 12, 8, 5, 13, 11, 15, 18

Evaluation

Evaluation will come out from the combination of: (1) work developed on the areas in the module, (2) attendance to lectures and participation in class and labs, and (3) a final exam.

The course consists of three parts: Theory, problem solving and Practice. The theory represents 50% of the final grade for the course, problem solving the 20% and the remaining 30% associated to practice. It is imperative that the average of the three tests will be at least 5.0 to pass the course. In addition, you need a minimum grade of 5.0 in each part of the course that has to approve it.

The evaluation of the theory will be a final exam. the day and time allocated for this purpose are indicated in the course of planning sheets, located in Virtual Campus .

The evaluation of the problem consists of two parts:
a) On the issue of assigned work, you must:
a1) Make a written work (10 pages long format IEEE- standard sections: abstract, introduction, related work,

role-specific content, results, conclusions, future work and references).
a2) A defense of (oral presentation of 20 minutes, 10 minutes for questions) the day and time allocated for this purpose in Virtual Campus.

Assistance and punctuality in all practice sessions is mandatory for all group members to overcome them.
To pass the practices required to have attended ALL practice sessions, its correct functioning, verification by the teacher responsible, answer the teacher's questions individually and presentation deadlines in a written report detailing:

• Target practice
• Description of approach and practice
• Description of the procedures used in functional
• Description of problems encountered during the realization of the practice and the solutions found
• Conclusions extracted from implementation of the practice
• Fixer configuration and testing

All those students who, having followed the ongoing assessment, have not achieved the minimum required to pass the course will have the option to do a test recovery. The date of this examination recovery will be published on the Virtual Campus course schedule.

Evaluation activities

Title	Weighting	Hours	ECTS	Learning outcomes
Exam	35%	2	0.08	2, 1, 5, 18
Lab work	35%	1	0.04	3, 1, 12, 16, 14, 6, 17, 8, 9, 5, 13, 11, 18
Presentation work	30%	1.5	0.06	2, 3, 1, 7, 10, 12, 16, 17, 6, 14, 8, 4, 5, 13, 11, 18

Bibliography

BOOKS

- Andrew S. Tanenbaum, "Computer Networks", 3ª Edición P.H. 1996.
- Grama, A. Gupta, G. Karypis, and V. Kumar,"Introduction to Parallel Computing, 2nd Ed. Addison-Wesley, 2003.
- Rajkumar Buyya, "High Performance Cluster Computing: Programming and Applications", PH, 1999.
- G. Coulouris, J. Dollimore and T. Kinderg, "Sistemas Distribuidos: Conceptos y Diseño", Addison-Wesley, 3ª Ed. 2001.
- Bell, Charles; Kindahl, Mats; Thalmann, Lars. "MySQL High Availability". O'Reilly, 2010.
- Chang, Fay, et al. "Bigtable: A Distributed Storage System for Structured Data." OSDI, 2006
- Dewitt, David, and Jim Gray. "Parallel Database Systems: The Future of High Performance Database Processing." Communications of the ACM 35, no. 6 (1992): 85-98
- Schwartz, Baron; Zaitsev, Peter; Tkachenko, Vadim; Zawodny, Jeremy D.; Lentz, Arjen; Balling, Derek J. "High Performance MySQL", O'Reilly, 2008.
- Taniar, David; Leung, Clement H.C.; Rahayu, Wenny; Goel, Sushant. "High Performance Parallel Processing and Grid Databases". Wiley, 2008.
- White, Tom. "Hadoop, the definitive Guide", O'Reilly, 2011.
- Ian Foster,Carl Kesselman. The grid: blueprint for a new computing infrastructure. Morgan-Kaufmann 2004.
- Mark Dowd, John McDonald, Justin Schuh. The Art of Software Security Assessment. Addison-Wesley 2007.
- Rickard Oberg. Mastering RMI: Developing Enterprise Applications in Java and EJB. John Wiley & Sons. 2001.

Websites recommended:

https://cv.uab.cat/