Universitat Autònoma de Barcelona

2022/2023

Applications of Coding Theory

Code: 105074
ECTS Credits: 6

Degree	Type	Year	Semester
2502441 Computer Engineering	OT	4	1

Contact

Name: Mercè Villanueva Gay
Email: merce.villanueva@uab.cat

Use of Languages

Principal working language: catalan (cat)
Some groups entirely in English: No
Some groups entirely in Catalan: Yes
Some groups entirely in Spanish: No

Teachers

Joaquim Borges Ayats
Cristina Fernandez Cordoba

Prerequisites

There are no prerequisites. However, students should have either a good mathematical level and be familiar with the concepts of fundamental algebra, or have passed the subjects "Informació i Seguretat" and "Fonaments de Tecnologies de la Informació".

Objectives and Contextualisation

The course is focused on coding theory and its applications into the real world. The coding theory is the study of methods for efficient and accurate transfer of information from one place to another. It deals with the problem of detecting and correcting transmission errors caused by noise on the channel. In distributed storage systems, coding theory also offers solutions, to improve hard disk failure tolerance, which are much more efficient than replication-based ones.

This course allows us to build the underground of connections by developing the "treball final de grau" (TFG) related to this topic and /or continuing a postgraduate related studies. It contemplates

the possibility of assuming this subject and the TFG
simultaneously.

Competences

- Acquire personal work habits.
- Acquire thinking habits.
- Capacity to design, develop, evaluate and ensure the accessibility, ergonomics, usability and security of computer systems, services and applications, as well as of the information that they manage.
- Capacity to design, develop, select and evaluate computer applications and systems, ensuring reliability, security and quality, in accordance with ethical principles, and applicable standards and legislation.
- Have the capacity to conceive, draft, organise, plan, develop and sign projects in the field of computer engineering for the conception, development and exploitation of computer systems, services and applications.
- Have the capacity to select, deploy, integrate and manage information systems that satisfy the needs of an organisation, identifying the cost and quality criteria.

Learning Outcomes

1. Design computer solutions that integrate accessibility and security needs in a distributed system.
2. Design, develop, select and evaluate applications, ensuring their reliability and security.
3. Design, develop, select and evaluate computer systems, ensuring their reliability, security and quality.
4. Develop a capacity for analysis, synthesis and prospection.
5. Identify the main attacks that a computer system can receive, as well as the possible protection and detection methods and the application of security policies to avoid damage to the system or minimise the repercussions.
6. Incorporate distributed information treatment systems in an organisation in order to increase operative capacity.
7. Work independently.

Content

1. Polynomials and finite fields.

    1.1. Rings of integers Z and Z/p.

    1.2. Ring of polynomials over Z/p.

    1.3. Finite fields GF(p^n)

1. Linear codes over finite fields.

    2.1. Introduction to coding theory.

2.2.
Generator matrices and equivalent codes
2.3.
Orthogonal codes and syndrome decoding
2.4.
Hamming codes

1.
Cyclic
codes over finite fields.

3.1.
Introduction to cyclic codes
3.2.
Generator polynomial and matrix
3.3.
Parity check polynomial and matrix
3.4.
Coding and decoding

1.
Algebraic
codes. BCH and RS codes.

4.1.
Introduction and general definitions
4.2.
Encoding an algebraic code
4.3.
Decoding an algebraic code
4.4.
BCH and RS codes
4.5.
Correcting errors and/or erasures

1.
Applications
of error correcting codes

5.1.
Error correcting codes in QR, Blu-ray, DVD.
5.2.
Error correcting codes in the transmission of
information.

5.3.
Error correcting codes applied to distributed storage.

5.4.
Cryptography based on error correcting codes.

5.5.
Error correcting codes applied to watermarking and steganography.

5.6.
Error correcting codes used in quantum computation.

Methodology

The methodology applied to the student work will combine the attended lectures, resolution of examples, practicum, and a short public talk about a specific subject previously approved. During the sessions, different concepts will be introduced and the resolution of exercises will be proposed to be solved by the students. The practicum proposals will be guided and will be validated by answering some questions. Campus Virtual will be used for communication between lecturers and students (material, updates, announcements, etc.).

Different activities will be conducted during the course:

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			
Practicum	12	0.48	4, 1, 2, 3, 5, 6, 7
Theoretical and practical classes / lectures	38	1.52	4, 1, 2, 3, 5, 6, 7
Type: Supervised			
Practicum supervising	6	0.24	4, 7
Report and oral presentation supervising or Tutoring for exam	6	0.24	4, 7
Tutoring and consultation	5	0.2	4, 7
Type: Autonomous			
Preparing exercices and practicum	35	1.4	4, 1, 2, 3, 5, 6, 7

Preparing oral presentation and/or exam	40	1.6	4, 1, 2, 3, 5, 6, 7

Assessment

Continuous-assessment
dates will be published on Campus Virtual. Specific programming may change when necessary. Any such modification will always be communicated to students through Campus Virtual, which is the usual communication platform between lecturers and students.

The
final evaluation will take into account the portfolio delivered by the students, the attendance and participation in class, and the short oral presentations, as follows:

1. Attendance
   and active participation. At least 80% of the lectures must be attended. Absences might be compensated with a home-work after agreement with the teacher. Mark: 10%.
2. Exercice
   resolutions. This is an individual task. As part of continuous assessment, short exercises must be solved. Some will be compulsory, others will be optional. Mark: 25%.
3. Practical
   activities. Depending on the number of students, it will be an individual task or in groups of two people. These practical activities will be performed by using computers. Mark: 25%.
4. Written
   work and oral presentation, and/or final exam, depending on the number of students and their profile. This is an individual task. It consists of an exam or delivering a written word and oral presentation about a specific topic. The choice of the topic will be discussed and agreed upon in the class, selecting topics from a list provided by the faculty staff or by the students themselves. In addition, the presenter will propose an exercise or questionnaire that the other students will have to answer. On the other hand, the other students in the audience must ask questions (at least one for each talk) during the presentations. A preliminary list of tentative topics are the ones described in chapter 5. Mark: 40%.

Notwithstanding
other disciplinary measures deemed appropriate, and in accordance with the academic regulations in force, assessment activities will receive a zero whenever a student commits academic irregularities that may alter such assessment. Assessment activities graded in this way and by this procedure will not be re-assessable. If passing the assessment activity or activities in question is required to pass the subject, the awarding of a zero for disciplinary measures will also entail a direct fail for the

subject, with no opportunity to re-assess this in the same academic year. Irregularities contemplated in this procedure include, among others

- the total or partial copying of an evaluation activity;
- allowing others to copy;
- presenting group work that has not been done entirely by the members of the group;
- presenting any materials prepared by a third party as one's own work, even if these materials are translations or adaptations, including work that is not original or exclusively that of the student;

An overall grade of 5 or higheris required to pass the subject. A "non-assessable" grade cannot be assigned to students who have participated in more than 50% of the exercises and practicum activities or have delivered the oral presentation. No special treatment will be given to students who have completed the course in the previous academic year. In order to pass the course with honours, the final grade must be a 9.0 or higher. Because the number of students with this distinction cannot exceed 5% of the number of students enrolled in the course, this distinction will be awarded to whoever has the highest final grade.

It is important to bear in mind that no assessment activities will be permitted for any student at a different date or time to that established, unless for justified causes duly advised before the activity and with the lecturer's previous consent. In all other cases, if an activity has not been carried out, this cannot be re-assessed.

In the case of exercise resolutions and practical activities, a review may be requested after the date of the activity, allowing students to review the activity with the lecturer. In this context, students may discuss the activity grade awarded by the lecturers responsible for the subject. If students do not take part in this review, no further opportunity will be made available.

To consult the academic regulations approved by the Governing Council

of the UAB, please follow this link:
http://webs2002.uab.es/afers_academics/info_ac/0041.htm

Assessment Activities

Title	Weighting	Hours	ECTS	Learning Outcomes
Attendance and active participation	10	0	0	4, 1, 2, 3, 5, 6, 7
Exercise resolution	25	3	0.12	4, 1, 2, 3, 5, 6, 7
Practical activities	25	3	0.12	4, 1, 2, 3, 5, 6, 7
Written work and oral presentation and/or exam	40	2	0.08	4, 1, 2, 3, 5, 6, 7

Bibliography

- C. H. Bennett, P. Shor, "Quantum Information Theory", IEEE Trans. Inf. Theory, vol. 44, n.6, pp. 2724-2742, 1998.
- D. J. Bernstein, J. Buchmann, E. Dahmen (Eds.), Post-Quantum Cryptography, Springer-Verlag, 2009.
- Thomas M. Cover and Joy A. Thomas (1991). Elements of Information Theory, John Wiley & Sons, Inc.
- K. Gracie and M.-H. Hamon, "Turbo and turbo-like codes: Principles and applications", IEEE Proceedings of in Telecommunications, vol. 95, pp: 1228 - 1254, 2000.
- K. J. Horadam, Hadamard Matrices and Their Applications, Princeton University Press, 2007.
- Robert J. McEliece, The Theory of Information and Coding, Addison-Wesley Publishing Co., 1977.
- J. Rifà and Ll. Huguet, Comunicación Digital, Masson Ed. 1991.
- P. Shor, "Algorithms for Quantum Computation: Discrete Logarithm and Factoring", Proceedings 35-th Annual Symposium on Foundations of Computer Science, pp. 124-134, 1994.
- Mc. Williams-Sloane: The Theory of Error-Correcting Codes. North-Holland Publishing Company. Amsterdam-N.Y.-Oxford. 1978-1996.

Software

The practical activities will be perform by using SageMath. https://www.sagemath.org/
SageMath is a free open-source mathematics software system licensed under the GPL. It builds on top of many existing open-source packages: NumPy, SciPy, matplotlib, Sympy, Maxima, GAP, FLINT, R and many more. Access their combined power through a common, Python-based language or directly via interfaces or wrappers. Since version 9.0 released in January 2020, SageMath is using Python 3.