



1. UNIDADE CURRICULAR (COM ECTS)

CURRICULAR UNIT (WITH ECTS)

PENSAMENTO ALGORÍTMICO E APLICAÇÕES
ALGORITHMIC THINKING AND APPLICATIONS

6 ECTS

2. DESIGNAÇÃO DO CICLO DE ESTUDOS EM QUE SE INSERE A UNIDADE CURRICULAR (COM SEMESTRE E ANO LETIVO)

STUDY CYCLE TO WHICH THE CURRICULAR UNIT BELONGS (WITH ACADEMIC SEMESTER AND SCHOLAR YEAR)

Mestrado em Ciências da Comunicação / 2º Semestre / 2025-2026

Master in Communication Studies / 2nd Semester / 2025-2026

3. DOCENTE(S) DA UNIDADE CURRICULAR

ACADEMIC STAFF

Gonçalo Valadão

4. CARGA LETIVA NA UNIDADE CURRICULAR

WEEKLY TEACHING HOURS

24 horas

24 hours

5. OBJETIVOS DE APRENDIZAGEM (CONHECIMENTOS, APTIDÕES E COMPETÊNCIAS A DESENVOLVER PELOS ESTUDANTES)

LEARNING OUTCOMES OF THE CURRICULAR UNIT

Objetivos:

No final desta unidade curricular o aluno deverá ser capaz de:

- Compreender a importância do pensamento algorítmico.
- Adquirir conhecimentos fundamentais em algoritmia e capacidade de pensamento algorítmico, usando a linguagem de programação Python para implementação em aplicações relevantes na área da comunicação.

Competências:

1. Conhecer noções fundamentais de algoritmia e programação.
2. Saber analisar problemas e desenvolver soluções algorítmicas usando pseudocódigo.
3. Ter conhecimentos elementares da linguagem de programação Python.
4. Saber traduzir um algoritmo escrito em pseudocódigo numa implementação utilizando a linguagem Python.



5. Compreender e saber utilizar estruturas de dados e algoritmos elementares em aprendizagem automática.
6. Ter presentes questões de ética imanentes à utilização de algoritmos.

Objectives:

At the end of this course unit, the student should be able to:

- *Understand the importance of studying and developing algorithmic thinking.*
- *Acquire fundamental knowledge in algorithms and algorithmic thinking skills, using the Python programming language for implementation in relevant applications in the field of communication.*

Competences:

1. *Understand fundamental concepts of algorithms and programming.*
2. *Know how to analyze problems and develop algorithmic solutions using pseudo-code.*
3. *Have elementary knowledge of the Python programming language.*
4. *Know how to translate an algorithm written in pseudo-code into an implementation using the Python language.*
5. *Understand and know how to use data structures and elementary algorithms in machine learning.*
6. *Address ethical issues inherent in the use of algorithms.*

6. CONTEÚDOS PROGRAMÁTICOS

SYLLABUS

1. Introdução ao pensamento algorítmico.
2. Pseudo-código como linguagem intermédia de representação algorítmica.
3. Introdução à linguagem de programação Python.
4. Tipos de dados e controle de fluxo.
5. Funções, âmbito, abstração e recursividade.
6. Introdução aos *large language models*. Considerações de ética.
7. Incorporação programática de *large language models* em soluções algorítmicas para problemas práticos na área da comunicação.

1. *Introduction to algorithmic thinking.*
2. *Pseudo-code as an intermediate language for algorithmic representation.*
3. *Introduction to the Python programming language.*
4. *Data types and flow control.*
5. *Functions, scope, abstraction, and recursion.*



6. *Introduction to large language models. Ethical considerations.*
7. *Programmatic incorporation of large language models into algorithmic solutions for practical problems in the field of communication.*

7. METODOLOGIA DE ENSINO (AVALIAÇÃO INCLUÍDA) TEACHING METHODOLOGIES (INCLUDING EVALUATION)

Metodologia

A metodologia de ensino basear-se-á em aulas teórico-práticas. Para a prática concreta de programação, serão usados computadores em sala de aula e em todas as aulas. Não obstante a ênfase prática, o enquadramento teórico será sempre devidamente salvaguardado. Os tópicos do programa serão ilustrados com diversos exemplos de aplicação prática a diversos problemas relevantes na área da comunicação. A capacidade crítica e reflexiva será estimulada e valorizada através de debate frequente, e através de apresentações de trabalhos realizadas por grupos de alunos em sala de aula.

Avaliação

A avaliação é contínua e inclui os seguintes elementos, com a ponderação respetiva:

Elemento de avaliação	% da classificação final
Conjuntos de problemas	50%
Trabalho(s) de grupo	50%
Total	100%

Qualquer evidência de plágio será punida com a atribuição de zero ao elemento de avaliação que tenha utilizado indevidamente textos de terceiros.

Methodology

The teaching methodology will be based on theoretical-practical classes. For the practical programming aspect, computers will be used in the classroom for all sessions. However, despite the practical emphasis, the theoretical framework will always be duly safeguarded. The program topics will be illustrated with various examples of practical applications to relevant problems in the field of communication. Critical and reflective abilities will be stimulated and valued through frequent debate and presentations of work conducted by student groups in the classroom.

Evaluation

Evaluation is continuous and includes the following components with respective weightings:

Assessment component	% of final mark
Problem Sets	50%
Group Work(s)	50%
Total	100%



Any evidence of plagiarism shall be penalized with a mark of zero (0). Plagiarism is the improper use of a text and/or ideas that are not one's own.

8. BIBLIOGRAFIA PRINCIPAL

MAIN BIBLIOGRAPHY

- Cormen, T. H. et al. (2022), Introduction to Algorithms, fourth edition, MIT Press.
- Guttag, John V. (2021). Introduction to Computation and Programming Using Python, third edition: With Application to Computational Modeling and Understanding Data. MIT Press.
- Downey, Alan B. (2022), Elements of Data Science, Green Tea Press.
- Riscutia, Vlad, (2023). Large Language Models at Work: Enhancing Software Systems with Language Models.
- Wentworth, Peter, Elkner, Jeffrey, Downey, Allen B., Meyers, Chris (2020). How to Think Like a Computer Scientist: Learning with Python 3 Documentation. Release 3rd Edition. Self-published.
- Bottou, Léon et al. (2013). Counterfactual Reasoning and Learning Systems: The Example of Computational Advertising. Journal of Machine Learning Discipline. International Journal of Communication, 13, 23.
- Creel, Kathleen and Hellman, Deborah, The Algorithmic Leviathan: Arbitrariness, Fairness, and Opportunity in Algorithmic Decision Making Systems (February 15, 2021). Virginia Public Law and Legal Theory Research Paper No. 2021-13, Available at SSRN: <https://ssrn.com/abstract=3786377>.
- Barger, Robert N. (2008). Computer Ethics: A Case-Based Approach. Cambridge University Press.