

Caracterização da Unidade Curricular / Characterization of the Curricular Unit

Identificação do Curso / Course Identification: Licenciatura em Engenharia Informática / Computer Science Engineering

Coordenador(a) de curso / Course Coordinator: Sérgio Sargo Lopes, Doutor

Designação da Unidade Curricular (UC) / Title of Curricular Unit (CU): Algoritmos e Modelos de Programação / Algorithms and Programming Models

Área científica da UC / CU Scientific Area: Ciências Informáticas / Computer Science

Semestre / Semester: 1º

Número de créditos ECTS / Number of ECTS credits: 6

Horas de trabalho / Working hours: 150

Carga horária por tipologia de horas / Workload by type of hours: TP: 22,5h; PL: 22,5h; OT: 6h; O: 9h

Carga letiva semanal / Weekly letive charge: 4,5h

Docente responsável / Responsible professor: Jorge Manuel Simões, Doutor

Membros da equipa de docência / Academic staff members: Jorge Manuel Simões, Doutor

Introdução:

Pretende-se que na unidade curricular de Algoritmos e Modelos de Programação os alunos adquiram as competências necessárias para a resolução algorítmica de problemas, conhecendo as várias formas de representação de algoritmos, e usem linguagens de programação para codificação desses algoritmos. A unidade curricular de Algoritmos e Modelos de Programação insere-se na área científica de Ciências Informáticas.

Introduction:

The goal of the course unit *Algorithms and Programming Models* is for students to acquire the necessary skills for the algorithmic problem-solving process, understanding various forms of algorithm representation, and using programming languages to code these algorithms. The *Algorithms and Programming Models* course unit falls under the scientific domain of Computer Science.

Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):

No final da unidade curricular os alunos deverão estar aptos a desenvolver um raciocínio lógico, em termos de conceitos e técnicas de programação, de forma a serem capazes de conceber algoritmos para a resolução de problemas práticos de pequena e média complexidade. Os alunos deverão desenvolver as capacidades necessárias para a aplicação dos algoritmos concebidos em programas de computador concretos recorrendo à linguagem Python.

Intended learning outcomes (knowledge, skills and competences to be developed by the students):

By the end of the course unit, students should be able to develop logical reasoning in terms of programming concepts and techniques, enabling them to design algorithms for solving practical problems of small to medium complexity. Students should also develop the necessary skills to implement the designed algorithms in concrete computer programs using the Python language.

Pré-requisitos e facilitadores de aprendizagem:

Conhecimentos de informática na ótica de utilizador. Conhecimentos de matemática ao nível do ensino secundário.

Prerequisites and learning facilitators:

Basic computer skills and a solid understanding of mathematics at the high school level.

Conteúdos programáticos:

1. Introdução à algoritmia
 - 1.1. Fundamentos e conceitos
 - 1.2. Definição de algoritmo
 - 1.3. Representação de algoritmos
 - 1.4. Decomposição algorítmica (top-down)
2. Conceitos de programação
 - 2.1. Conceção de um programa
 - 2.2. Execução de um programa
 - 2.3. Compiladores e interpretadores
 - 2.4. Linguagens de programação
 - 2.5. Introdução à linguagem de programação Python
 - 2.6. Instalação e configuração do ambiente de programação
3. Programação em Python
 - 3.1. Tipos de dados, variáveis e constantes
 - 3.2. Operadores e expressões
 - 3.4. Instruções condicionais
 - 3.5. Instruções cíclicas
 - 3.4. Subprogramas
 - 3.5. Variáveis indexadas: vetores e matrizes
 - 3.6. Módulos e funções
 - 3.7. Variáveis indexadas
 - 3.8. Ficheiros
 - 3.9. Bibliotecas

Syllabus:

1. Introduction to algorithms
 - 1.1. Fundamentals and concepts
 - 1.2. Algorithm Definition
 - 1.3. Representation of algorithms
 - 1.4. Algorithmic decomposition (top-down)
2. Programming concepts
 - 2.1. Designing a program
 - 2.2. Executing a program
 - 2.3. Compilers and interpreters
 - 2.4. Programming languages
 - 2.5. Introduction to the Python programming language
 - 2.6. Installation and configuration of the programming environment
3. Python Programming
 - 3.1. Data types, variables and constants
 - 3.2. Operators and expressions
 - 3.4. Conditional statements
 - 3.5. Cyclic instructions
 - 3.4. Subprograms
 - 3.5. Indexed variables: vectors and matrices
 - 3.6. Modules and functions
 - 3.7. Indexed variables
 - 3.8. Files
 - 3.9. Libraries

Demonstração da coerência dos conteúdos programáticos com os objetivos da unidade curricular:

A unidade curricular prevê que os alunos adquiram conhecimentos de algoritmia e desenvolvam capacidades de resolução algorítmica de problemas diversos. Como suporte a essa resolução é usada a linguagem de programação Python para além de outras ferramentas para a elaboração de fluxogramas e desenvolvimento low-code. A linguagem Python é usada de acordo com o paradigma de programação procedimental, destinando-se a possibilitar aos alunos a concretização concreta das soluções algorítmicas dos problemas e exercícios propostos.

Evidence of the syllabus coherence with the curricular unit's intended learning outcomes:

The course unit aims for students to acquire knowledge of algorithms and develop problem-solving skills through algorithmic approaches. To support this resolution, the Python programming language is used, along with other tools for creating flowcharts and low-code development. Python is employed according to the procedural

programming paradigm, enabling students to concretely implement the algorithmic solutions to the problems and exercises proposed.

Metodologia de ensino (avaliação incluída):

A unidade curricular é constituída por uma componente teórico-prática onde são abordados os temas que constam do programa (metodologia expositiva combinada com sala de aula invertida) e por uma componente prática onde, através da realização de trabalhos, são aplicados os conhecimentos teóricos adquiridos (metodologia ativa com técnicas de aprendizagem baseada em problemas). Cada trabalho prático será executado durante uma sessão de prática-laboratorial de 1,5h. Para além dos trabalhos práticos haverá uma prova escrita no final do semestre.

A avaliação da unidade curricular, na época de avaliação por frequência será efetuada da forma seguinte:

- a) Trabalhos Práticos (TP1 e TP2), a realizar nas aulas práticas laboratoriais, com pesos de 20% e de 30%.
- b) Prova Escrita (PE), com um peso de 40%.
- c) Assiduidade e Participação nas atividades propostas (AP), com um peso de 10%.

Classificação Final = 20% TP1 + 30% TP2 + 40% PE + 10% AP

A classificação final, nas épocas de avaliação por exame final, recurso, especial e conclusão de curso, será efetuada da forma seguinte:

Classificação Final = 20% TP1 + 30% TP2 + 50% PE

Datas previstas para as avaliações:

- Trabalho prático 1 (TP1): 11 novembro de 2025 (turma diurna) e 12 novembro de 2025 (turma pós-laboral).
- Trabalho prático 2 (TP2): 13 janeiro de 2026 (turma diurna) e 14 janeiro de 2026 (turma pós-laboral).
- Prova Escrita: A marcar pela Direção da Escola Superior de Ciência e Tecnologia.

Lective methodologies (including students' assessment):

The course unit consists of a theoretical-practical component, in which the topics included in the syllabus are addressed (expository methodology combined with a flipped classroom approach), and a practical component, where theoretical knowledge is applied through assignments (active methodology using problem-based learning techniques). Each practical assignment will be carried out during a 1.5-hour laboratory-practice session. In addition to the practical assignments, there will be a written exam at the end of the semester.

The assessment of the course unit, during the continuous assessment period, will be carried out as follows:

- a) Practical Assignments (TP1 and TP2), to be completed in the laboratory practice classes, with weightings of 20% and 30%.
- b) Written Exam (PE), with a weighting of 40%.
- c) Attendance and Participation in the proposed activities (AP), with a weighting of 10%.

Final Grade = 20% TP1 + 30% TP2 + 40% PE + 10% AP

The final grade, in the assessment periods corresponding to the final exam (normal, appeal and special periods), will be calculated as follows:

Final Grade = 20% TP1 + 30% TP2 + 50% PE

Planned assessment dates:

- Practical Assignment 1 (TP1): November 11, 2025 (day class) and November 12, 2025 (evening class).
- Practical Assignment 2 (TP2): January 13, 2026 (day class) and January 14, 2026 (evening class).
- Written Exam: To be scheduled by the Board of the School of Science and Technology.

Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:

A exposição de conhecimentos teóricos é complementada com aulas de prática laboratorial onde os alunos, através de exercícios práticos, resolvem projetos que permitem a aplicação desses conceitos. Para fomentar as atividades práticas é aplicada também a abordagem da sala de aula invertida. Alguns dos projetos práticos são usados para a avaliação dos alunos. Os alunos participam ainda em outras atividades práticas para estimular o pensamento computacional recorrendo a diferentes plataformas para desenvolvimento de fluxogramas e de aplicações low-code. Estas abordagens permitem que os alunos pratiquem continuamente os conhecimentos adquiridos ao longo do semestre. As provas escritas, realizadas em ambiente web, contribuem para a avaliação sumativa dos conhecimentos adquiridos.

Evidence of the coherence between the teaching methodologies and the intended learning outcomes:

The delivery of theoretical knowledge is complemented by laboratory practice classes in which students, through practical exercises, solve projects that enable the application of these concepts. To further promote practical activities, the flipped classroom approach is also applied. Some of the practical projects are used for student assessment. Students also take part in other practical activities designed to stimulate computational thinking, using different platforms for developing flowcharts and low-code applications. These approaches allow students to continuously practice the knowledge acquired throughout the semester. Written exams, carried out in a web-based environment, contribute to the summative assessment of the knowledge acquired.

Bibliografia principal / Main bibliography:

- Portela, F., Pereira, T. (2023). Introdução à Algoritmia e Programação com Python, FCA, ISBN: 978-972-722-931-4
- Sobral, S. (2024). Introdução à Programação usando Python, 2ª edição, Edições Sílabo, ISBN: 978-989-561-387-8
- Sobral, S. (2024). Introdução à Programação usando Python – Exercícios Resolvidos, Edições Sílabo, ISBN: 978-989-561-386-1

Simões, J., Santos, M. (1996), Introdução à Programação, Ed. Catepse, ISBN: 972-97083-0-4, Cota: 004.43/SIMj/INT ISPGaya

Bibliografia complementar / Supplementary bibliography:

Carvalho, A. (2021). Práticas de Python: Algoritmia e Programação, FCA, 978-972-722-918-5

Vasconcelos, J. (2015). Python: Algoritmia e Programação Web, FCA, ISBN: 978-972-722-813-3

Costa, E. (2015). Programação em Python: Fundamentos e Resolução de Problemas, FCA, ISBN: 978-972-722-816-4

Links úteis / Useful links:

<https://www.geeksforgeeks.org/dsa/introduction-to-algorithms/>

<https://www.w3schools.com/python/>

<https://docs.python.org/3/tutorial/index.html>

<https://www.codecademy.com/learn/learn-python-3>

<https://anandology.com/python-practice-book/>

Avaliação | Condições de Acesso à Época de Recurso:

De acordo com o regulamento de avaliação.

Assessment | Conditions of Access to the Season of Appeal:

According to the assessment regulation.

Avaliação | Condições de Melhoria de Classificação:

De acordo com o regulamento de avaliação.

Assessment | Classification Improvement Conditions:

According to the assessment regulation.

Recursos Didáticos:

Laboratório de computadores, videoprojetor, acesso Internet, LMS Moodle.

Didatics Resources:

Computers' lab, videoprojector, Internet access, LMS Moodle.

Idioma da Lecionação:

Português.

Teaching Language:

Portuguese.

Horário de Atendimento a Alunos:

Terça-feira, 17h30 – 18h30, mediante marcação previa por email (jsimoes@ispgaya.pt).

Student Service Hours:

Tuesday, 17h30 – 18h30 by appointment with the teacher by e-mail (jsimoes@ispgaya.pt).