instituto superior de engenharia de lisboa

Ficha de Unidade Curricular A3ES
Computação e Programação
Licenciatura em Engenharia Química e Biológica

2025-26

1. Designação da unidade curricular

[4413] Computação e Programação / Computation and Programming

2. Sigla da área científica em que se insere	CC, CEE
3. Duração	Unidade Curricular Semestral
4. Horas de trabalho	108h 00m
5. Horas de contacto	Total: 30h 00m das quais T: 10h 30m | P: 19h 30m
6. % Horas de contacto a distância	Sem horas de contacto à distância
7. ECTS	4
8. Docente responsável e respetiva carga letiva na Unidade Curricular	[1294] Sérgio Jorge Pereira da Costa | Horas Previstas: 1440 horas
9. Outros docentes e respetivas cargas letivas na unidade curricular	[1259] José Valério do Nascimento Palmeira | Horas Previstas: 45 horas
10. Objetivos de aprendizagem e a sua compatibilidade com o método de ensino (conhecimentos, aptidões e competências a desenvolver pelos estudantes).	Esta unidade curricular (UC) tem por objetivo dotar os alunos de competências na área da computação e programação, nomeadamente nos domínios da algoritmização e da programação de forma a adquirir conhecimentos de suporte e de competências fundamentais para a compreensão, desenvolvimento e execução de problemas de engenharia com diferentes graus de complexidade, através do desenvolvimento de rotinas em Python. Os alunos deverão compreender e aplicar os conceitos associados à Computação e Programação, principalmente no que respeita ao pensamento estruturado.

10. Intended Learning objectives and their compatibility with the teaching method (knowledge, skills and competences by the students).

This curricular unit (UC) objective is to provide students with skills in the area of computer science and computer engineering programming, namely in algorithms and programming in order to acquire knowledge and skills for understanding, develop and solve engineering problems with different degrees of complexity through the development of routines in Python. Students should be able to understand and apply the concepts associated with Computing and Programming especially with regard to structured thinking.

11. Conteúdos programáticos

1. Introdução à computação e programação. Introdução ao Python.

2. Variáveis. Atribuição, declaração, inicialização e representação de variáveis. Funções internas (built-in functions). Representação gráfica. Ficheiros script (py-files).

3. Definição e interpretação de problemas. Variáveis de entrada e saída.

4. Algoritmos, estruturas e programação estruturada (Top-Down e Bottom-Up). Pseudocódigo. Operadores relacionais e lógicos.

5. Atribuições condicionais (if/elif/else). Ciclos (while e for).

6. Funções definidas pelo utilizador: definição, construção e execução.

7. Computação e Programação simbólica.

11. Syllabus

1. Introduction to Computer Science and Programming. Introduction to Python.

2. Variables. Variable assignment, declaration, initialization and representation. Built-in functions. Graphical representation. Script files (py-files).

3. Problem definition and interpretation. Input and output variables.

4. Algorithms, structures and structured programming design (top-down and bottom-up). Pseudocode. Logical and relational operators.

5. Conditional statements (if/elif/else). Loops (while and for).

6. User-defined functions: define, build and execute.

7. Symbolic computation and programming.

12. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular

O programa da UC visa familiarizar os alunos com técnicas de resolução de problemas, criação e desenvolvimento de algoritmos, agilização do pensamento estruturado de forma a abordar a resolução de problemas teóricos/práticos de forma simplificada.

A utilização do Python serve para dotar os alunos de uma ferramenta/linguagem que servirá para o desenvolvimento de rotinas para esta e outras UCs que funcionem em simultâneo ou mais tarde no curso de Engenharia Química e Biológica.

Após a introdução (item 1) que contextualiza e realça a importância da computação e programação, as aulas focar-se-ão na linguagem de programação (itens 2-7). Os alunos deverão usar os conceitos apresentados para desenvolver formas de resolução de problemas.

No final da UC o aluno deverá estar apto a compreender os mais diversos problemas ao nível da engenharia, nomeadamente química e biológica e apresentar soluções a diferentes níveis.

12. Evidence of the syllabus coherence with the curricular unit´s intended learning outcomes

The syllabus content of this UC aims for students to be familiar with problem solving techniques, algorithm creation/development, as well as thinking in a structured way to address theoretical/practical problem solving in a simple way.

The use of Python provides all students with a programming language/tool to be used for routine development for this and other UC lectured simultaneously or later on in Chemical and Biological Engineering course.

After introduction (item 1), where the main goal is to give context and emphasize the importance of computation and programming, classes will focus on the programming language (items 2- 7). Students should make use of presented concepts to develop ways of problem solving.

At the end of the UC, all students should be able to understand the most diverse problems in engineering, particularly in the chemical and biological aspects and present solutions at different levels.

13. Metodologias de ensino e de aprendizagem específicas da unidade curricular articuladas com o modelo pedagógico

Os alunos terão aulas teóricas (AT) para apresentação de conceitos e aulas de prática laboratorial (APL) para aplicação dos mesmos e resolução de exercícios com exemplos teóricos/reais de Engenharia Química e Biológica, recorrendo a software específico. As APL são consideradas pedagogicamente fundamentais de frequência obrigatória.

13. Teaching and learning methodologies specific to the curricular unit articulated with the pedagogical model

Students will have theoretical classes (AT) focused on the presentation of concepts and practical classes (APL) for concept application and solving theoretical/real examples on Chemical and Biological Engineering, by means of specific software. The APL are considered pedagogically fundamental and mandatory.

14. Avaliação

A avaliação de conhecimentos é feita de forma distribuída, sem exame final, com a resolução de três trabalhos (TE1, TE2 e TE3), onde TE1 e TE2 são de resolução em período letivo em sala de aula, sem nota mínima e que podem conter total ou parcialmente o programa da UC.

O TE3, considerado pedagogicamente fundamental e de carácter obrigatório, poderá ser sujeito a discussão, e tem nota mínima de 8,00.

A Nota Final (NF) é obtida pela média ponderada dos três trabalhos, conforme a fórmula: NF=0,30x(TE1+TE2)/2+0,70xTE3, TE3 >= 8,00 ; NF >= 9,50.

A avaliação de conhecimentos não contempla a realização de exames, exames parciais ou a repetição de trabalhos.

14. Assessment

The evaluation assessment is distributed with no exam and consists of three practical assignments (WT1, WT2 and WT3), where WT1 and WT2 are to be solved during lecturing period in classroom, with no minimum grade and that can contain partial or full UC programme.

T3 is considered pedagogically fundamental and mandatory, can be subject to discussion,, and with a minimum grade of 8.00.

The Final Grade (FG) is obtained by the weighted average of the three practical assignments, by the formula: FG=0.30x(WT1+WT2)/2+0.70xWT3, WT3 >= 8.00; FG >= 9.50.

Knowledge assessment does not include exams, partial exams, or practical assignment repetition.

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

As aulas desta UC apresentam as duas componentes, teórica (T) e prática laboratorial (PL), com pesos iguais para que os alunos tenham não só tempo em sala de aula para apreender os conceitos teóricos, mas também tempo de os aplicar, praticar e discutir em tempo real, em contexto de sala de aula, os resultados com colegas e docente.

As aulas serão dadas em blocos de 1,5h e os horários e horas semanais são adaptados e adequados às matérias lecionadas e às necessidades variáveis entre teoria e prática, ou seja, o balanço semanal de horas em sala entre T e PL é variável e depende da dificuldade das matérias, das necessidades de apreensão dos alunos e da proximidade dos períodos de avaliação.

As aulas teóricas são expositivas, recorrendo à projeção de *slides* que servem para apresentar as diferentes matérias contidas no programa da UC. Estes materiais serão fornecidos aos alunos para que estes possam aplicar/adaptar aos diferentes problemas práticos apresentados nas aulas laboratoriais.

As aulas práticas são lecionadas em laboratório de informática, que disponibiliza um computador pessoal por aluno com o *software* necessário à resolução dos problemas apresentados. Serão de cariz expositivo sempre que necessário e de resolução de problemas práticos.

Em determinadas aulas, os alunos terão que executar dois trabalhos práticos (TE1 e TE2) que visem a resolução de problemas de engenharia. Com estes trabalhos pretende-se avaliar o desenvolvimento e consolidação dos conhecimentos dos alunos. Terão que conceber, desenvolver e executar rotinas de programação que permitam aferir da consolidação dos conteúdos estudados. Todos os trabalhos terão de ser entregues em formato digital e submetidos na página Moodle da UC e deverão incluir todas as rotinas desenvolvidas, bem como outros ficheiros solicitados.

O Trabalho considerado pedagogicamente fundamental (TE3), deverá seguir as indicações dos trabalhos TE1 e TE2, bem como a entrega de um guião/relatório com a explicação das operações realizadas e as respostas às perguntas incluídas no enunciado do TE3.

No final do semestre o TE3 será avaliado e, se necessário, alvo de discussão.

Durante o semestre letivo os alunos terão acesso a suporte físico e/ou digital dos diferentes itens abordados no programa da UC e deverão estar aptos a resolver a avaliação de forma individual.

15. Evidence of the teaching methodologies coherence with the curricular unit´s intended learning outcomes

The classes of this UC have two components, theoretical (T) and practical (PL), with equal weights so that students not only have time in the classroom to apprehend the theoretical concepts, but also time to apply, practice and discuss in real time the results with colleagues and teacher, in a classroom context. The classes will be given in blocks of 1.5 hours and the weekly schedules and hours are adapted to the subjects taught and the needs between theory and practice, that is, the weekly balance of hours in classroom between T and PL may vary and depends on the difficulty of the subjects, students? needs and the proximity of the evaluation periods.

Theoretical classes are mainly expositive, by means of slide projection used to present the different subjects of the curricular unit programme. These materials will be provided to the students, in order to their application/adaptation to different practical problems, to be presented in practical classes. The practical classes are taught in a computer lab, which should provide a personal computer per student with the necessary software to solve the problems presented. They will be of an expositive nature whenever necessary and practical problem solving. In certain classes, students will have to carry out two practical assignments (WT1 and WT2) aiming to solve an engineering problem. These assignments intend to evaluate the development and consolidation of students' knowledge. Students will have to create, develop, and execute programming routines that, along with other tools, allowing to gauge the consolidation of the studied contents.

All assignments (including WT3) will have to be presented in digital format and submitted in the Moodle page of the UC and should include all the routines developed and other requested documents. WT3 must also include, additionally to WT1 and WT2 criteria, a guide/report explaining the operations performed and the answers to the questions included in the assignment statement.

At the end of the semester all assignments are evaluated and, if necessary, WT3 could be subject to discussion.

During the semester students will have access to physical and/or digital support of the different items addressed in the UC program and should be able to solve the assessment in an individual way.

16. Bibliografia de consulta/existência obrigatória

1. Matthes, E. (2023). Python Crash Course: A Hands-On, Project-Based Introduction to Programming (3rd ed.). No Starch Press.

2. Turk, I. (2023). Python Programming: for Engineers and Scientists, CreateSpace Independent Publishing Platform.

3. Gries, P., Campbell, J., & Montojo, J. (2018). Practical Programming: An Introduction to Computer Science Using Python 3.6 (3rd. ed.). Pragmatic Bookshelf.

4. Lutz, M. (2013). Learning Python (5th ed.). O'Reilly Media.

5. Van Hattem, R. (2023). Mastering Python. Packt Publishing Limited.

17. Observações

Unidade Curricular Obrigatória

Data de aprovação em CTC: 2025-07-30

Data de aprovação em CP: 2025-07-29