

## Ficha de Unidade Curricular – (Versão A3ES 2018-2023)

### 1. Caracterização da Unidade Curricular.

**1.1. Designação da unidade curricular (1.000 carateres).**

Programação / Programming

**1.2. Sigla da área científica em que se insere (100 carateres).**

IC

**1.3. Duração<sup>1</sup> (100 carateres).**

Semestral

**1.4. Horas de trabalho<sup>2</sup> (100 carateres).**

162 h

**1.5. Horas de contacto<sup>3</sup> (100 carateres).**

Total - 67,5 h

T - 37,5 h

TP - 18 h

PL - 12 h

**1.6. ECTS (100 carateres).**

6

**1.7. Observações<sup>4</sup> (1.000 carateres).**

Comum com outros cursos

**1.7. Remarks (1.000 carateres).**

Common with other courses

**2. Docente responsável e respetiva carga letiva na Unidade Curricular (preencher o nome completo) (1.000 carateres).**

Pedro Alexandre de Seia e Cunha Ribeiro Pereira

**3. Outros docentes e respetivas cargas letivas na unidade curricular (1.000 carateres).**

**4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes). (1.000 carateres).**

Os estudantes que terminam com sucesso esta unidade curricular serão capazes de:

1. Demonstrar o conhecimento sobre as construções elementares de uma linguagem de programação multi-paradigma e com tipificação estática.
2. Construir pequenos programas que resolvam problemas simples descritos em linguagem natural.
3. Escrever, testar, depurar e corrigir pequenos programas.
4. Produzir documentação técnica onde se justificam as decisões tomadas nos programas construídos.
5. Utilizar eficazmente ferramentas para desenvolver programas e para elaborar relatórios.

**4. Intended learning outcomes (knowledge, skills and competences to be developed by the students). (1.000 characters).**

Students who successfully complete this course unit be able to:

1. Demonstrate know-how on the basic constructs of a statically typed and multi-paradigm programming language.
2. Build small programs aimed at solving simple problems described in natural language.
3. Write, test, debug and fix small programs.
4. Write reports justifying decisions made while building computer programs.
5. Make effective use of the tools to produce small computer programs and to write technical reports.

**5. Conteúdos programáticos (1.000 carateres).**

- I. Valores, tipos e variáveis; Mutabilidade e imutabilidade; Expressões; Entrada/Saída de dados; Decisão

binária e decisão múltipla; Funções; Ativação de funções e passagem de parâmetros; Funções como parâmetros; Expressões lambda.

II. Construções de controlo de execução no estilo imperativo e no estilo funcional; Recursão e co-recursão.

III. Coleções de dados; Operações sobre coleções: filter, map e fold; Algoritmos elementares sobre coleções.

IV. Tipos primitivos e definição de tipos; Composição de dados homogênea (coleções); Composição de dados heterogênea (objetos).

**5. Syllabus (1.000 characters).**

I. Values, Types and variables; Mutability and immutability; Expressions, Input/Output; Binary and multiple decision instructions. Functions; Function activation and parameters passing; Functions as parameters; Lambda expressions.

II. Execution control constructs: in both imperative and functional programming styles; Recursion and corecursion.

III. Data collections; Basic operations on collections: filter, map, fold; Elementary algorithms on data collections.

IV. Primitive types and type definition; Homogeneous data composition (collections); heterogeneous data composition (objects).

**6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular (1.000 carateres).**

Esta unidade curricular representa para a maioria dos alunos um primeiro contacto com a programação, que se pretende motivador sem descurar o formalismo e o rigor, sendo a base da formação de software do curso. São introduzidos conceitos e vocabulário fundamental da programação imperativa e funcional.

Os objetivos 1, 2 e 3 são alcançados através dos itens I até IV do conteúdo programático.

O desenvolvimento de programas durante as sessões práticas permite atingir os objetivos 4 e 5.

**6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes (1.000 characters).**

This curricular unit is for most students a first contact with programming. Must be motivator without neglecting the formality and rigor and is the basis of software training course. Are introduced the fundamental concepts and vocabulary of imperative and functional programming.

Goals 1, 2 and 3 are achieved through items I to IV of the syllabus.

The development of programs during practical sessions makes it possible to achieve goals 4 and 5 of learning outcomes.

**7. Metodologias de ensino (avaliação incluída) (1.000 carateres).**

Previstas 30 aulas durante o semestre (15 de 3 horas e 15 de 1,5 horas). Quatro aulas de 3 horas são de prática laboratorial e as restantes teóricas e teórico-práticas. As aulas destinam-se à apresentação dos temas e de exemplos práticos de aplicação. Os tópicos principais são ainda explorados através da realização de trabalhos práticos para desenvolver pequenos programas.

Os resultados da aprendizagem (1) e (2) são avaliados individualmente através do teste final escrito e das fichas realizadas durante o semestre. Durante o acompanhamento dos trabalhos de grupo realizados nas aulas práticas são avaliados os resultados da aprendizagem (3) e (5). Os resultados da aprendizagem (4) e (5) são avaliados na discussão final dos trabalhos de grupo.

Nota final corresponde a 40% do teste final, 40% dos trabalhos e 20% das fichas realizadas nas aulas.

**7. Teaching methodologies (including assessment) (1.000 characters).**

Planned during the semester in 30 lectures (15 of 3 hours and 15 of 1.5 hours). Four lectures of 3 hours are laboratory classes. The lectures are intended for presentation of topics and practical examples. The main topics are further explored through practical work to develop small programs.

Learning outcomes (1) and (2) are individually assessed through short written tests during the semester and final test. During the monitoring of group work performed in class practices are assessed learning outcomes (3) and (5). The learning outcomes (4) and (5) are evaluated in the final discussion of work group.

Final grade corresponds to 40% of the final test, 40% of the practical works and 20% of the evaluation sheets made in class.

**8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular (3.000 carateres).**

Nas sessões teóricas e teórico-práticas são apresentados os mecanismos básicos da linguagem de programação e são construídos, testados e corrigidos pequenos programas com a participação dos alunos. Para realizar os diversos trabalhos práticos são usadas ferramentas de complexidade gradualmente crescente (linha de comando, editor simples, ambiente integrado, depurador de erros) e são escritos relatórios que fundamentam as decisões tomadas.

**8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes**  
(3.000 characters).

In theoretical and practical sessions are presented the basic mechanisms of the language and are built, tested and corrected small programs with student participation. To achieve its various practical work, are used gradually increasing complexity tools (command line, simple editor, integrated environment, debugger) and are written reports that support the decisions made.

**9. Bibliografia de consulta/existência obrigatória (1.000 caracteres).**

I. Galata; J. Howard; D. Lucas and E Shapiro, *Kotlin Apprentice: Beginning Programming with Kotlin*, first edition, Razeware LLC, 2018. ISBN-13: 978-1942878506; ISBN-10: 1942878508

---

<sup>1</sup> Anual, semestral, trimestral, ...

<sup>2</sup> Número total de horas de trabalho.

<sup>3</sup> Discriminadas por tipo de metodologia adotado (T - Ensino teórico; TP - Ensino teórico-prático; PL - Ensino prático e laboratorial; TC - Trabalho de campo; S - Seminário; E - Estágio; OT - Orientação tutorial; O - Outro).

<sup>4</sup> Assinalar sempre que a unidade curricular seja optativa.