

Ficha de Unidade Curricular – Programação I

1. Caracterização da Unidade Curricular.

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

Programação I/Programming I

Assinado por: **CARLOS EDUARDO DE MENESES RIBEIRO**

Num. de Identificação: 06906457

Data: 2023.09.20 17:10:21+01'00'

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

IC

1.3. **Duração¹ (100 carateres).**

Semestral

Assinado por: **Pedro Manuel de Almeida**

Carvalho Vieira

Num. de Identificação: 10302000

Data: 2023.09.21 09:09:17 +0100

1.4. **Horas de trabalho² (100 carateres).**

160

Assinado por: **MÁRIO PEREIRA VÉSTIAS**

Num. de Identificação: 09139891

Data: 2023.09.21 11:44:24+01'00'

1.5. **Horas de contacto³ (100 carateres).**

67,5h (T – 45h; PL- 22,5h)

Aprovado pelo CP. A Presidente do CP

Assinado por: **SANDRA MARIA DA SILVA**

FIGUEIREDO ALEIXO

Num. de Identificação: 07789825

Data: 2023.09.22 23:42:22 +0100

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

6

1.7. **Observações⁴ (1.000 carateres).**

UC comum a LEETC e LEIRT



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

Common to LEETC and LEIRT

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

Jorge Manuel Rodrigues Martins Pião (67,5h)

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

Manuel Fernandes Carvalho (135h); Nuno António Afonso Cunha de Oliveira (67,5h); Mário Justiniano Morais Pinheiro (135h)

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 os mecanismos básicos das linguagens procedimentais.
2. Construir programas de pequena complexidade em linguagem C a partir da sua descrição em linguagem natural.
3. Usar ferramentas para construir, testar e corrigir pequenos programas.

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

Students who successfully complete this course will be able to:

1. Demonstrate knowledge of the basic mechanisms of procedural languages.
2. Build small complexity programs in C language from its natural language description.
3. Use tools to build, test and debug small programs.

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

- I. Linguagens de programação versus linguagens naturais. Sintaxe e semântica. Algoritmos.
- II. Tipos de dados elementares. Valores, operadores e expressões. Operações de entrada/saída de dados. Variáveis.
- III. Instruções de controlo de fluxo: decisão simples e múltipla; ciclos. Funções. Passagem de parâmetros. Alcance e tempo de vida de variáveis.

- IV. Tipos estruturados: *arrays*; *strings*; estruturas; *arrays* multidimensionais.
- V. Algoritmos básicos de pesquisa e ordenação.
- VI. Introdução aos ponteiros. Acesso a ficheiros.
- VII. Introdução à programação modular.

5. Syllabus (1.000 characters).

- I. Programming languages versus natural languages. Syntax and semantics. Algorithms.
- II. Basic data types. Values, operators and expressions. Input /output operations. Variables.
- III. Flow control instructions: simple and multiple decision; cycles. Functions. Parameters passing. Scope and lifetime of variables.
- IV. Structured types: arrays; strings; structures; multidimensional arrays.
- V. Basic algorithms for searching and sorting.
- VI. Introduction to pointers. File accessing.
- VII. Introduction to modular programming.

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

Esta unidade curricular representa para a maioria dos estudantes o 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 procedimental (pontos I, II, III e IV do conteúdo programático) concretizados na linguagem C. Os algoritmos de pesquisa e ordenação e o acesso a ficheiros (V, VI) são usados para praticar os conceitos transmitidos. O conceito de ponteiro e a introdução à programação modular (pontos VI e VII) preparam estudos posteriores na área da programação e da arquitetura de computadores.

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

This course is for most students a first contact with programming, with motivational purposes, without neglecting formalism and rigor, being the basis of software training course. The fundamental concepts and vocabulary of procedural programming (Sections I, II, III and IV of the syllabus) are presented using the C language. Searching and sorting algorithms and files access aim to practice the transmitted concepts. Pointers and an introduction to modular programming are introduced in (section VI and VII) prepare new topics in the area of programming and computer architecture.

7. Metodologias de ensino (avaliação incluída) (1.000 caracteres).

Aulas teóricas (45h), com apresentação dos temas, exposição dos conceitos e exemplos de aplicação, complementadas com aulas de prática laboratorial (22,5h) para realização de exercícios de consolidação.

A avaliação tem as componentes escrita (E) e prática (P).

A componente E pode ser realizada em dois testes (T1 e T2) ou em exame final. T1 é realizado na fase letiva. Na época normal de exame, o estudante pode optar por T2, se obteve o mínimo em T1, ou exame. Na época de recurso, pode optar por realizar T2 ou exame.

A nota mínima em cada teste escrito é de 8 valores, sendo a nota da componente E obtida da média dos testes ou da nota de exame, com o mínimo de 9,5 valores.

A componente P é baseada em três trabalhos práticos, realizados em grupo, com o conhecimento de cada estudante avaliado durante a discussão do trabalho, sendo atribuídas classificações individuais.

Cada trabalho tem como nota mínima, incluindo a discussão, de 9,5 valores.

A nota final (C) é obtida através da fórmula: $C = E*0,5 + P*0,5$.

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

Theoretical classes (45h), with presentation of themes, exposition of concepts and application examples, complemented with laboratory practice classes (22.5h) for the development of consolidation exercises.

The assessment has written (E) and practical (P) components.

Component E can be performed in two tests (T1 and T2) or in a final exam. T1 is carried out in the class phase. In the normal exam period, student could opt for T2, if he obtained the minimum in T1, or exam. At the time of appeal, students can opt or T2 or exam.

Minimum score in each written test is 8, with the E component score being obtained by average the tests or the exam score, with a minimum of 9.5.

The P component is based on 3 practical assignments, carried out in groups, with each student's knowledge evaluated during the discussion of the work, with individual classifications being attributed. Each work has a minimum of 9.5, including the discussion.

The final grade (C) is obtained using the formula: $C=E*0.5+P*0.5$, in a scale of 0 to 20.

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

Os conteúdos programáticos são expostos nas aulas teóricas, complementando uma apresentação interativa das matérias com a realização, pelos estudantes, de pequenos exercícios de consolidação das mesmas.

Os objetivos de aprendizagem são desenvolvidos na realização dos trabalhos de grupo.

São efetuadas aulas práticas de acompanhamento dos trabalhos de grupo, que complementam a avaliação, na discussão final, da globalidade dos objetivos de aprendizagem.

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

The syllabus contents are exposed in the theoretical classes, complementing an interactive presentation of the subjects with the realization, by the students, of small consolidation exercises.

Learning objectives are developed in carrying out group work.

Practical classes are carried out to support the group work, which complement the assessment, in the final discussion, of the learning objectives.

9. Bibliografia de consulta/existência obrigatória (1.000 carateres).

K.N.King, "C Programming: a Modern Approach, 2Ed.", W. W. Norton & Company, 2008.

B. Kernighan, D. Ritchie, *The C Programming Language*, 2ª ed., Prentice Hall, 1988.

¹ Anual, semestral, trimestral, ...

² Número total de horas de trabalho.

³ 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).

⁴ Assinalar sempre que a unidade curricular seja optativa.