

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

1 Caracterização da Unidade Curricular.

1.1 Designação da unidade curricular (1.000 carateres).

Programação de Computadores (PC - 3877)

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

EE

1.3 Duração (100 carateres).

Semestral

1.4 Horas de trabalho (100 carateres).

148,5h

1.5 Horas de contacto (100 carateres).

67,5h; TP:45h; PL:22,5h.

1.6 ECTS (100 carateres).

5,5

1.7 Observações (1.000 carateres).

1.7 Remarks (1.000 carateres).

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

Fernando Joaquim Ganhão Pereira

9h

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

José Carlos de Ponte Ribeiro

9h

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

- Fomentar, explorar e desenvolver, a capacidade de raciocínio e abstração;
- Aplicar conceitos básicos de programação em linguagens de alto nível;
- Utilizar a linguagem de programação C na resolução de problemas afins à Engenharia Eletrotécnica.

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

- to promote, to explore and to develop, the capacity of reasoning and abstraction;
- Apply basic programming concepts in high level languages;
- To use the C programming language in solving problems related to Electrotechnical Engineering.

5. Conteúdos programáticos (1.000 carateres).

Capítulo 1 – Noções Gerais

- Arquitetura básica de um computador
 - CPU, RAM, ROM dispositivos de entrada e saída.
- Representação de informação num computador
 - Sistemas de numeração (Binário, Octal e Hexadecimal)
 - Representação de números inteiros e reais
 - Representação de caracteres
- Sistemas Operativos e Linguagens de programação
 - Linguagens de baixo nível e linguagens de alto nível
 - Linguagens compiladas e interpretadas
 - Funções do Sistema Operativo

Capítulo 2 – Algoritmia

- Linguagens de descrição de algoritmos
 - Pseudo-código
 - Fluxogramas
- Conceção de Algoritmos
 - Abordagem top-down

Capítulo 3 – Linguagem de programação C

- Conceitos fundamentais
 - Estrutura de um programa
 - Programa “*hello_world*”
 - A função *main()*
 - Bloco de instruções
 - Comentários
 - Indentação
 - Palavras reservadas
 - Utilização de Bibliotecas
- Variáveis e constantes
 - Declaração de variáveis
 - Constantes (*const* e *#define*)
 - Atribuição
 - Regras para identificadores
- Tipos Primitivos
 - Tipos inteiros: *int short, long, unsigned*
 - formatação de leitura e escrita (*scanf* e *printf*)
 - funções standard com inteiros
 - Tipos reais: *float* e *double*
 - formatação de leitura e escrita
 - funções standard com reais
 - Tipo *char*
 - caracteres especiais
 - funções standard com caracteres
 - *Cast*
- Operadores e expressões
 - Operadores Aritméticos

- Operadores relacionais
- Operadores lógicos
- Operadores binários
- Operadores ?
- Operadores de incremento e decremento
- Operadores de atribuição composta
- Precedência de operadores
- *Mecanismos de controlo de fluxo*
 - Mecanismos de seleção
 - *if...else*
 - *switch*
- Mecanismos de repetição
 - *while*
 - *do...while*
 - *for*
 - Instruções *break* e *continue*
- Vetores e Apontadores
 - Vetores unidimensionais
 - Vetores multi-dimensionais
 - Apontadores
 - Memória dinâmica
 - Cadeias de caracteres ("*Strings*")
 - Funções para manipulação de *strings*
- Funções
 - Cabeçalho e corpo de uma função
 - Parâmetros
 - Passagem de parâmetros por valor e referência
 - Valor de retorno. O tipo *void*
 - Funções matemática standard
 - *Return* e *exit*
 - Parâmetros do programa (*argc* e *argv*)
 - Módulos
 - *Protótipos de funções*
 - *Implementação de um novo módulo*
 - *Ficheiro de definições (.h)*
 - *Ficheiro de implementação (.c)*
 - "*Linkagem*"
 - Funções Recursivas
- O pré-processador
 - Definição de símbolos
 - Definição de macros
 - Compilação condicional
 - Símbolos como opções do compilador
- Estruturas
 - Definição de estruturas

- Declaração de variáveis do tipo estrutura
- Acesso a membros de uma estrutura
- Ficheiros
 - Abertura e fecho de ficheiros
 - Leitura e escrita de ficheiros

Acesso sequencial e direto

5. Syllabus (1.000 characters).

Chapter 1 - The Basics

Basic architecture of a computer

the CPU, RAM, ROM, input and output devices.

Representation of information in a computer

the Number Systems (Binary, Octal and Hexadecimal)

the representation of integers and real numbers

representation of characters

Operating Systems and Programming Languages

the low-level languages and high level languages

the compiled and interpreted languages

functions of Operating System

Chapter 2 - Algorithms

Description languages algorithms

pseudo-code

the Flowcharts

Design of Algorithms

the top-down approach

Chapter 3 - Programming Language C

Key Concepts

the structure of a program

program "hello_world"

the function main ()

block instruction

comments

the Indentation

the reserved words

use of Libraries

Variables and Constants

declaration of Variables

the constants (const and #define)

the Assignment

rules for identifiers

Primitive Types

the integer types: int short, long, unsigned

formatting for reading and writing (scanf and printf)

standard functions on integers

the real types: float and double

- formatting reading and writing

- standard functions with real

- the char type:

- special characters

- functions with standard characters

- the Cast

- Operators and Expressions

- the Arithmetic Operators

- the Relational Operators

- the Logical Operators

- the Binary Operators

- the operator ?

- the increment and decrement operators

- the compound assignment operators

- the Operator Precedence

- Flow control mechanisms

- The mechanisms of selection

- if ... else

- switch

- Mechanisms repeat

- while ...

- do ... while

- for

- break and continue

- Vectors and Pointers

- one-dimensional vectors

- multi-dimensional vectors

- pointers

- the dynamic memory

- the strings ("Strings")

- functions for manipulating strings

- Functions

- the header and body of a function

- the Parameters

- the Passing parameters by value and reference

- the return value. The type void

- the standard mathematical functions

- return the exit

- program parameters (argc and argv)

- Modules

- function prototypes

- implementing a new module

- definition file (. h)

- implementation File (.c)

- "Linking"

Recursive Functions

The preprocessor

definition of symbols

definition of macros

the Conditional Compilation

symbols such as the compiler options

Structures

definition of structures

the declaration of variables of type structure

access to members of a structure

Files

opening and closing the file

the reading and writing of files

the direct and sequential access

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

- Em todo o programa curricular de Programação de Computadores faz-se uma abordagem dos problemas no âmbito matemático, em especial no campo da matemática aplicada à eletrotécnica de forma a fomentar e explorar a capacidade de raciocínio e abstração;
- Todas as elaborações de conteúdos de suporte à Unidade Curricular e implementação dos programas em linguagem C, aplicam conceitos básicos e fundamentais para a estruturação e conceção de algoritmos que visam a sua posterior programação em linguagens de alto nível;
- Todo o estudo feito na Unidade Curricular é realizado de modo a seguir uma abordagem imperativa segundo os "Computing Curricula for Computer Science" para a linguagem de programação C e segue todas as "Curricula Recommendations" da ACM-Association for Computing Machinery;
- Os Conteúdos programáticos estão de acordo com a ACM-Association for Computing Machinery e visam seguir todas as suas recomendações.

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

- In the entire curriculum of Computer Programming, an approach to the problems in the mathematical scope is made, especially in the field of mathematics applied to the electrotechnology in order to foment and to explore the capacity of reasoning and abstraction;
- All the elaborations of contents of support to the Curricular Unit and implementation of the programs in C language, apply basic concepts and fundamental for the structuring and design of algorithms that aim at its later programming in high level languages;
- All the study done in the Curricular Unit is carried out in order to follow an imperative approach according to the Computing Curricula for Computer Science for programming language C and follows all the "Curricula Recommendations" of the ACM-Association for Computing Machinery;
- The Programmatic Content is in agreement with the ACM-Association for Computing Machinery and aims to follow all its recommendations.

7. Metodologias de ensino (avaliação incluída) (1000 caracteres).

- Abordagem Imperativa de Programação de Computadores, complementada com o ensino da Linguagem de Programação C.

Avaliação de Conhecimentos:

- Componente Teórica (NT) : avaliação através de prova escrita no final do semestre (Nota mínima de 9,5 valores).
- Componente Prática (NP) : avaliação através de trabalhos práticos, individuais e/ou em grupo, com discussão individual no final do semestre (Nota mínima de 9,5 valores).
- Nota Final (NF): Média aritmética das duas componentes: $NF = \frac{NT + NP}{2}$

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

- Imperative Approach of Computer Programming, complemented with the teaching of the Programming Language C.

Knowledge Assessment:

Theoretical Component (TC): evaluation through written test at the end of the semester (minimum mark of 9.5 points).

Practical Component (PC): evaluation through practical, individual and / or group work, with individual discussion at the end of the semester (minimum mark of 9.5 points).

Final Note (FN): Arithmetic average of two components: $FN=(TC+PC)/2$.

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

Com a metodologia seguida até aqui, tem-se verificado que os objetivos propostos para a aprendizagem nos diferentes níveis (conhecimentos, aptidões e competências) são alcançados e desenvolvidos pelos estudantes no final da frequência da unidade curricular, permitindo-lhes a utilização destes conhecimentos na sua vida académica e profissional, quer dentro do âmbito da eletrotécnica e automação, quer em outras áreas do conhecimento em que o raciocínio e a abstração são fundamentais.

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

With the methodology followed so far, it has been verified that the proposed objectives for learning at different levels (knowledge, skills and competences) are achieved and developed by the students at the end of the course unit, allowing them to use this knowledge in their academic and professional life, both within the scope of electrotechnology and automation, or in other areas of knowledge where the rationale and abstraction are fundamental.

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

- *Fundamentos de Programação*, Marques de Sá, FCA - Editora de Informática

- *The C Programming Language - The ANSI edition*, Brian W. Kernighan e Dennis M. Ritchie, Prentice-Hall
- Material de apoio (C Reference Card, Exercícios, Programas, Exames Tipo) está publicado na área de Programação de Computadores que se encontra disponível na plataforma MOODLE do ISEL referente à Área Departamental (ADEEEA) em que a Unidade Curricular se insere.
- Sebenta constituída pelas folhas em PDF de toda a matéria Teórica referente ao conteúdo programático da Unidade Curricular , disponível na plataforma MOODLE do ISEL e na Área Departamental (ADEEEA) em que a Unidade Curricular se insere.

Sebenta constituída pelas folhas em PDF de toda a matéria Prática referente ao conteúdo programático da Unidade Curricular , disponível na plataforma MOODLE do ISEL e na Área Departamental (ADEEEA) em que a Unidade Curricular se insere.