

---

## 1. Caracterização da Unidade Curricular

### 1.1 Designação

[3885] Estruturas de Dados e Algoritmos / Data Structure and Algorithms

### 1.2 Sigla da área científica em que se insere

EE

### 1.3 Duração

Unidade Curricular Semestral

### 1.4 Horas de trabalho

148h 30m

### 1.5 Horas de contacto

Total: 67h 30m das quais T: 45h 00m | P: 22h 30m

### 1.6 ECTS

5.5

### 1.7 Observações

Unidade Curricular Obrigatória

---

## 2. Docente responsável

[1210] Fernando Joaquim Ganhão Pereira

---

## 3. Docentes e respetivas cargas letivas na unidade curricular

[1210] Fernando Joaquim Ganhão Pereira | Horas Previstas: 22.5 horas

[1567] José Carlos de Ponte Ribeiro | Horas Previstas: 292.5 horas

[2071] Rui Manuel Luzia Guerreiro | Horas Previstas: 45 horas

---

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

Adquirir conhecimentos avançados em estruturas de dados, algoritmos e tipos de dados abstratos. Comparar as propriedades de diferentes algoritmos. Dominar a programação orientada por objetos. Dominar a utilização da linguagem de programação C++. Conceber programas com estruturas de dados e algoritmos complexos.

---

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

To acquire advance knowledge in data structures, algorithms and abstract data types. To compare different algorithms properties To get well acquainted with OOP. To be familiarized with C++ programming language. To implement programs with complex algorithms and data structures.

---

#### 5. Conteúdos programáticos

- Programação estruturada em C++.
- Programação com classes em C++.
- Análise de algoritmos e sua complexidade.
- Algoritmos de ordenação e pesquisa em tabelas ou listas.
- Definição de árvores. Árvores binárias. Algoritmos de inserção, pesquisa e remoção.
- Tabelas de dispersão.
- Definição de grafos. Matrizes de adjacência.
- Algoritmos de procura e emparelhamento.
- Algoritmo do Caminho mais curto e Fluxo máximo.

---

#### 5. Syllabus

- Structured programming with C++.
- Classes in C++.
- Principles of algorithm analysis.
- Sort and search algorithms.
- Binary Trees. Insertion, search and delete algorithms.
- Graphs. Adjacency matrix.
- Search and matching algorithms.
- Shortest-Path Algorithms. Network Flow Problems

---

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

O estudo de algoritmos e estruturas de dados complexos são introduzidos com base num programa coerente de conteúdos associado à resolução de trabalho laboratorial suportado em programas desenvolvidos pelos estudantes com utilização dum ambiente de desenvolvimento de software, Visual Studio da Microsoft.

---

#### 6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes

The study of algorithms and complex data structures are introduced in a coherent set of topics associated with experimental work supported in the development of several programs written in C++ executed by the students with the help of Microsoft Visual Studio Integrated Development Environment.

---

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

A unidade curricular é ministrada em aulas teóricas e práticas laboratoriais.

A avaliação de conhecimentos é Distribuída com Exame Final, todas as componentes de avaliação são pedagogicamente fundamentais:

- Componente teórica (NT), avaliada por um exame escrito em época normal ou de recurso, com nota mínima de 9,50 valores;
  - Componente prática (NP), avaliada pela realização de 6 trabalhos práticos em grupo, com entrega de relatórios dos mesmos e uma discussão oral na última semana de aulas, com nota média mínima de 9,50 valores.
- A nota final (NF) é dada pela média aritmética das duas componentes,  $NF = (NT+NP)/2$ , nota final mínima de 9,50.

---

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

The curricular unit is taught in theoretical classes and laboratory practices.

The assessment is distributed with a final exam, all evaluation components are pedagogically fundamental:

- The theoretical component (NT), assessed by a written exam in the normal or appeal period, with minimum grade of 9.50;
- The practical component (NP), assessed by carrying out 6 practical group assignments with delivery reports and an oral discussion in the last week of classes, with a minimum average grade of 9.50 values.

The final grade (NF) is given by the arithmetic mean of the two components,  $NF = (NT+NP)/2$ , with minimum final grade of 9.50.

---

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

Os trabalhos experimentais servem para comprovar na prática que os conteúdos que são ministrados nas aulas teóricas são corretamente assimilados. Estes trabalhos são constituídos pelo desenvolvimento de pequenos programas em C++ que são encadeados para que o último trabalho seja um acumular de todos os anteriores.

---

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

Experimental work is used to show in practice that the topics ministered in theoretical lectures are well assimilated by students. These works are basically programs written in C++ linked together in a manner that the last work is a compilation of all the rest.



---

**9. Bibliografia de  
consulta/existência obrigatória**

- "Data Structures and Algorithm Analysis in C++", 2nd ed., Mark Allen Weiss, Addison-Wesley, 1999.
- "Algorithms in C++", Robert Sedgewick; Addison-Wesley, 1998.
- "Programação em C++", Conceitos Básicos e Algoritmos, P. Rodrigues, P. Pereira, M. Sousa, FCA - Editora de Informática, 1998.
- "Programação em C++", Algoritmos e Estruturas de Dados, P. Rodrigues, P. Pereira, M. Sousa, FCA - Editora de Informática, 2000.
- "The C++ Programming Language", 3rd ed., Bjarne Stroustrup, Addison-Wesley, 1997

---

**10. Data de aprovação em CTC** 2024-07-17

---

**11. Data de aprovação em CP** 2024-06-26