

---

**Ano Letivo** 2018-19

---

**Unidade Curricular** PROGRAMAÇÃO ORIENTADA POR OBJETOS

---

**Cursos** ENGENHARIA INFORMÁTICA (1.º ciclo)  
ENGENHARIA ELETRÓNICA E TELECOMUNICAÇÕES (Mestrado Integrado)

---

**Unidade Orgânica** Faculdade de Ciências e Tecnologia

---

**Código da Unidade Curricular** 140064323

---

**Área Científica** CIÊNCIA DE COMPUTADORES

---

**Sigla**

---

**Línguas de Aprendizagem** Português-PT

---

**Modalidade de ensino** Presencial diurno

---

**Docente Responsável** José Luís Valente de Oliveira

DOCENTE	TIPO DE AULA	TURMAS	TOTAL HORAS DE CONTACTO (*)
José Luís Valente de Oliveira	PL; T	T1; PL1	30T; 30PL
Docente A Contratar FCT 1	PL	PL2; PL3	60PL

\* Para turmas lecionadas conjuntamente, apenas é contabilizada a carga horária de uma delas.

ANO	PERÍODO DE FUNCIONAMENTO*	HORAS DE CONTACTO	HORAS TOTAIS DE TRABALHO	ECTS
2º	S1	30T; 30PL	168	6

\* A-Anual;S-Semestral;Q-Quadrimestral;T-Trimestral

#### Precedências

Sem precedências

#### Conhecimentos Prévios recomendados

PROGRAMAÇÃO IMPERATIVA

LABORATÓRIO DE PROGRAMAÇÃO

#### Objetivos de aprendizagem (conhecimentos, aptidões e competências)

No fim desta disciplina os alunos deverão ser capazes de entender e aplicar os princípios e as técnicas de programação orientada por objectos. Em particular, deverão ser capazes de gerar uma especificação UML recorrendo a padrões de desenho e implementar essa especificação na linguagem JAVA.

#### Conteúdos programáticos

1. Princípios e conceitos fundamentais da programação orientada por objetos.
2. Introdução modelação orientada por objetos e ao UML- Unified Modeling Language.
3. Tipos de dados abstractos.
4. Pilhas.
5. Filas.
6. Padrões de projeto fundamentais: Template Method, Strategy, Iterator.
7. A linguagem JAVA.

### **Metodologias de ensino (avaliação incluída)**

Método clássico de ensino e aprendizagem para as disciplinas científico-tecnológicas.

Aulas teóricas expositivas com recurso ao quadro e projector de vídeo.

Aulas práticas com tutoriais, problemas e mini-projectos de programação, incluindo actividades que vão desde a modelação, implementação e depuração.

Avaliação

Nota final = 0,6 Exame + 0,4 Avaliação Prática Arredondamentos só na nota final.

### **EXAMES**

Os exames consistem numa prova escrita com consulta de até 25 páginas A4 numa fonte de tamanho não inferior a 9. São admitidos a exame, independentemente da época, os estudantes cuja nota de avaliação prática seja  $\geq 7,5$  valores.

### **AVALIAÇÃO PRÁTICA**

Média ponderada de trabalhos práticos.

Os trabalhos são realizados em grupo, de inscrição obrigatória. A nota prática do grupo é convertida numa nota prática individual no momento da discussão dos trabalhos. A nota prática é, portanto, individual, estando dependente do desempenho de cada elemento do grupo.

---

### **Bibliografia principal**

[Referência principal]

Bruce Eckel. Thinking in Java, 4th edition, Prentice Hall, New Jersey, 2006, cf. <http://mindview.net/Books/TIJ4>

[Referência complementar sobre padrões de projeto]

Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. Design Pattern ? Elements of Reusable Object-Oriented Software, Addison-Wesley, 1995

[Referência complementar sobre UML] UML Quick Ref: <http://www.uml.org>

[Referência complementar sobre Estruturas de Dados]

Robert Sedgewick. Algorithms in Java, 3rd edition ? parts 1-4: Fundamentals, data structures, sorting, searching, Addison-Wesley, 2003

**Academic Year** 2018-19

**Course unit** OBJECT-ORIENTED PROGRAMMING

**Courses** INFORMATICS (COMPUTER SCIENCE) (1st Cycle)  
ELECTRONIC ENGINEERING AND TELECOMMUNICATIONS (Integrated Master's)

**Faculty / School** Faculdade de Ciências e Tecnologia

**Main Scientific Area** CIÊNCIA DE COMPUTADORES

**Acronym**

**Language of instruction** Português-PT

**Teaching/Learning modality** Daytime presential

**Coordinating teacher** José Luís Valente de Oliveira

Teaching staff	Type	Classes	Hours (*)
José Luís Valente de Oliveira	PL; T	T1; PL1	30T; 30PL
Docente A Contratar FCT 1	PL	PL2; PL3	60PL

\* For classes taught jointly, it is only accounted the workload of one.

#### Contact hours

T	TP	PL	TC	S	E	OT	O	Total
30	0	30	0	0	0	0	0	168

T - Theoretical; TP - Theoretical and practical ; PL - Practical and laboratorial; TC - Field Work; S - Seminar; E - Training; OT - Tutorial; O - Other

#### Pre-requisites

no pre-requisites

#### Prior knowledge and skills

Imperative Programming

Programming Lab

#### The students intended learning outcomes (knowledge, skills and competences)

1. Enumerate, describe, and justify the object-oriented concepts, principles, and techniques.
2. Application modeling using UML, emphasizing on architectural aspects (diagrams of classes and objects)
3. To Select and employ fundamental design patterns.
4. To use JAVA as a object-oriented programming language

#### Syllabus

1. Object-Oriented concepts and principles;
2. Introduction to Object Oriented Modelling and UML (Unified Modelling Language);
3. Abstract data types
4. Stacks.
5. Queues.
6. Fundamental Design Patterns : Template Method, Strategy, Iterator.
7. Object Oriented Programming using Java

#### Teaching methodologies (including evaluation)

Written examination: 60%; Pratical evaluation: 40%

### Main Bibliography

[Main reference]

Bruce Eckel. Thinking in Java, 4th edition, Prentice Hall, New Jersey, 2006, cf. <http://mindview.net/Books/TIJ4>

[Other optional references]

Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. Design Pattern ? Elements of Reusable Object-Oriented Software, Addison-Wesley, 1995

UML Quick Ref: <http://www.uml.org>

Robert Sedgewick. Algorithms in Java, 3rd edition ? parts 1-4: Fundamentals, data structures, sorting, searching, Addison-Wesley, 2003