UAlg
UNIVERSIDADE DO ALGARVE

English version at the end of this document

| | |
|---|---|
| **Ano Letivo** | 2018-19 |
| **Unidade Curricular** | COMPLEMENTOS DE PROGRAMAÇÃO |
| **Cursos** | SISTEMAS E TECNOLOGIAS DE INFORMAÇÃO |
| **Unidade Orgânica** | Instituto Superior de Engenharia |
| **Código da Unidade Curricular** | 18121006 |
| **Área Científica** | FORMAÇÃO TÉCNICA,CIÊNCIAS INFORMÁTICAS |
| **Sigla** | FT |
| **Línguas de Aprendizagem** | Português |
| **Modalidade de ensino** | Presencial |
| **Docente Responsável** | Pedro Jorge Sequeira Cardoso |

| DOCENTE | TIPO DE AULA | TURMAS | TOTAL HORAS DE CONTACTO (*) |
|---|---|---|---|
| Pedro Jorge Sequeira Cardoso | PL; TP | TP1; PL1 | 15TP; 45PL |

* 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 |
|---|---|---|---|---|
| 1º | S2 | 15TP; 45PL | 140 | 5 |

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

**Precedências**

Sem precedências

**Conhecimentos Prévios recomendados**

Conceitos de programação procedimental. Frequência da UC de Programação.

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

Proporcionar uma visão geral sobre os conceitos definidos em programação orientada a objetos (POO). Capacitar o aluno a oferecer soluções algorítmicas para problemas computacionais.

Utilizar uma linguagem de POO para implementação das soluções.

**Conteúdos programáticos**

Introdução ao paradigma da Programação Orientada a Objetos
1. Perspetiva histórica
2. Características da programação orientada a objetos (encapsulamento, partilha de comportamento, evolução)
3. Objetos (noção de Objeto e identidade; protocolo de utilização de um Objeto; mensagens e métodos; noção de estado e comportamento.
4. Linguagens baseadas em classes; instanciação de Objetos.
5. Hierarquias de Objetos: Mecanismos de herança; Reutilização de código; Sistemas reflexivos; Polimorfismo.
6. Interfaces: Separação entre interface e implementação de interfaces.
7. Relações entre Objetos: Extensão, generalização e especialização; Associação, agregação e composição; Coesão e acoplamento.
8. Aplicações (GUI, Ligações a Bases de Dados, Desenvolvimento de aplicações móveis)

**Metodologias de ensino (avaliação incluída)**

**Metodologias de ensino**

Aulas Teórico-Práticas: Método expositivo, orientado de acordo com um ensino baseado em problemas, com projeção e explicação dos objetivos e conteúdos correspondentes a cada semana.

Aulas Práticas: Prática laboratorial em computador. Resolução e codificação de problemas tipificados, selecionados em conformidade com o conteúdo teórico semanal.

**Avaliação**

Componentes da avaliação classificadas de 0-20 valores:

- Trabalho(s) prático(s) (TP)
- Prova escrita (PE):  Teste / Exame

Classificação mínima em cada uma das componentes 7 valores.

**Nota final:** 0.5 PE + 0.5 TP

De acordo com o n.º 3 do artigo 6.º do despacho reitoral RT 59/2015, de 28 de julho, nos cursos técnicos superiores profissionais, a inclusão do cumprimento do dever de assiduidade nos métodos de avaliação é obrigatória, nos seguintes termos:
a) Considera-se que um estudante cumpre o dever de assiduidade a uma UC, quando não exceda o número limite de faltas correspondente a 25% das horas de contacto previstas

**Bibliografia principal**

[1] Java 6 - Programação para Objetos. F. Mário Martins, FCA Editora.

[2] Fundamentos de Programação em Java 2, A. J. Mendes e M. J. Marcelino, FCA Editora.

[3] Introduction to Programming Using Java, Sixth Edition, D. J. Eck, http://math.hws.edu/javanotes/

[4] The Java Tutorials, http://download.oracle.com/javase/tutorial/

UAlg
UNIVERSIDADE DO ALGARVE

| | |
|---|---|
| **Academic Year** | 2018-19 |
| **Course unit** | PROGRAMMING II |
| **Courses** | SISTEMAS E TECNOLOGIAS DE INFORMAÇÃO |
| **Faculty / School** | Instituto Superior de Engenharia |
| **Main Scientific Area** | FORMAÇÃO TÉCNICA,CIÊNCIAS INFORMÁTICAS |
| **Acronym** | FT |
| **Language of instruction** | Portuguese |
| **Teaching/Learning modality** | Presential |
| **Coordinating teacher** | Pedro Jorge Sequeira Cardoso |

| Teaching staff | Type | Classes | Hours (*) |
|---|---|---|---|
| Pedro Jorge Sequeira Cardoso | PL; TP | TP1; PL1 | 15TP; 45PL |

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

**Contact hours**

| T | TP | PL | TC | S | E | OT | O | Total |
|---|---|---|---|---|---|---|---|---|
| 0 | 15 | 45 | 0 | 0 | 0 | 0 | 0 | 140 |

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**

Concepts of procedural programming. Frequency of Programming CU.

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

Provide an overview of concepts defined in object-oriented programming (OOP). Enable student to offer algorithmic solutions to computational problems.

Use an OOP language to implement programming solutions.

**Syllabus**

Introduction to the Object Oriented Programming Paradigm
1. Historical perspective
2. Characteristics of object-oriented programming (encapsulation, behavior sharing, evolution)
3. Objects (notion of object and identity, protocols of object's usage, messages and methods, notion of state and behavior).
4. Class-based languages; instantiation of objects.
5. Object Hierarchies: Inheritance mechanisms; Reuse of code; Reflection; Polymorphism.
6. Interfaces.
7. Relations between Objects: Extension, generalization and specialization; Association, aggregation and composition; Cohesion and coupling.
8. Applications (GUI, database connections)

**Teaching methodologies (including evaluation)**

Teaching methodologies

Theoretical-Practical classes: Expositive method, oriented according to a problem-based teaching, with projection and explanation of the objectives and contents corresponding to each week.

Practical classes: Laboratory practice in computer. Resolution and codification of typified problems, selected according to the theoretical weekly content.

Evaluation

Valuation components classified as 0-20 values:

- Practical assignments (TP)
- Written exam (PE): Test / Exam

Minimum classification in each of the components: 7 values.

Final grade: 0.5 PE + 0.5 TP

Pursuant to no. 3 of Article no. 6 of RT 59/2015, of July 28, in the professional higher technical courses, the inclusion of the fulfillment of the duty of assiduity in the methods of evaluation is obligatory, in the following terms:
a) It is considered that a student fulfills the duty of assiduity to a CU, when it does not exceed the limit of absences corresponding to 25% of the foreseen contact hours

**Main Bibliography**

[1] Java 6 - Programação para Objetos. F. Mário Martins, FCA Editora.

[2] Fundamentos de Programação em Java 2, A. J. Mendes e M. J. Marcelino, FCA Editora.

[3] Introduction to Programming Using Java, Sixth Edition, D. J. Eck, http://math.hws.edu/javanotes/

[4] The Java Tutorials, http://download.oracle.com/javase/tutorial/