
Ano Letivo 2021-22

Unidade Curricular TECNOLOGIAS DE MERCADO

Cursos TECNOLOGIAS INFORMÁTICAS

Unidade Orgânica Instituto Superior de Engenharia

Código da Unidade Curricular 19321005

Área Científica CIÊNCIAS INFORMÁTICAS

Sigla

Código CNAEF (3 dígitos) 481

Contributo para os Objetivos de Desenvolvimento Sustentável - ODS (Indicar até 3 objetivos) 4, 8, 9

Línguas de Aprendizagem Português

Modalidade de ensino

Presencial

Docente Responsável

Pedro Jorge Sequeira Cardoso

DOCENTE	TIPO DE AULA	TURMAS	TOTAL HORAS DE CONTACTO (*)
Docente A Contratar ISE 10	PL; TP	TP1; PL1	30TP; 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	S1	20TP; 55PL	182	7

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

Precedências

Sem precedências

Conhecimentos Prévios recomendados

- Conceitos básicos de OOP (herança, encapsulamento, estado dos objetos, comportamento dos objetos, polimorfismo, etc.)
- Ler e elaborar diagramas UML e classes e Objetos

Objetivos de aprendizagem (conhecimentos, aptidões e competências)

O principal objetivo desta unidade curricular da unidade curricular de Tecnologias de Mercado, é cimentar os conhecimentos de OOP tirando o máximo proveito das especificidades oferecidas pelos ambientes virtuais de execução Java.

No final da unidade curricular os estudantes devem ser capazes de:

- Identificar os diferentes tipos primitivos disponibilizados pela Standard Edition do Java;
 - Aplicar boas práticas de programação;
 - Implementar diferentes controlos de fluxo: Branching e Loops;
 - Consultar as definições dos diferentes tipos oferecidos pela Standard Edition usando documentação da Java API;
 - Definir os seus próprios tipos Java;
 - Tratar e definir exceções;
 - Ser capaz de elaborar soluções Object Oriented, desenhando diagramas de classes UML, implementando classes Java tirando partido de conceitos como herança, polimorfismo, contratos abstratos e tipos genéricos.
-

Conteúdos programáticos

Introdução

- Introdução à história do Java
- Código java vs byte codes vs Java Virtual Machine

Computação básica

- diferentes tipos Java
- operações sobre strings de caracteres
- convenções
- comentários uteis

Controlo de Fluxo:

- branching
- tipos primitivos e referência
- tipo primitivo boolean
- utilização de enumerados simples
- implementar loops (while, do, for)

Definição de classes e métodos

- criar objetos, classes e métodos
- diagramas de classes UML
- construtores

- métodos estáticos e definição de variáveis estáticas

Arrays

- Ordenação e pesquisa de arrays
- Arrays multidimensionais

Herança, Polimorfismo e Interfaces

- conceitos de polimorfismo e herança
- Definição de interfaces
- Dynamic Binding em Java
- Classes derivadas

Exceções

- tratamento de exceções
- Errors vs Exceptions vs Runtime Exceptions e exceções Checked e Unchecked

Estruturas de dados dinâmicas e genéricas

- instâncias de ArrayList
- Estruturas de dados simplesmente ou duplamente ligadas
- Iteradores

Metodologias de ensino (avaliação incluída)

No âmbito da UC serão utilizadas as seguintes metodologias:

- Exposição teórico-prática da matéria;
- Debate dos temas abordados e esclarecimento de dúvidas;
- Estímulo à participação, interação e dinâmica de grupo;
- Realização de trabalhos práticos para aplicação dos conhecimentos;

A classificação final será calculada da seguinte forma:

- Exame Final e mínimo de 9,5 valores (40%);
- Componente prática e avaliada por uma ficha de exercícios e um trabalho final (60%);
- A nota será atribuída após discussão com o docente, sobre todo o trabalho prático elaborado ao longo da unidade curricular;"

Bibliografia principal

Walter Savitch, Java: An introduction to problem solving & Programming. Pearson; 8ª edição

Academic Year 2021-22

Course unit

Courses COMPUTER TECHNOLOGIES

Faculty / School INSTITUTE OF ENGINEERING

Main Scientific Area

Acronym

CNAEF code (3 digits) 481

Contribution to Sustainable Development Goals - SGD (Designate up to 3 objectives) 4, 8, 9

Language of instruction Portuguese

Teaching/Learning modality Presencial

Coordinating teacher Pedro Jorge Sequeira Cardoso

Teaching staff	Type	Classes	Hours (*)
Docente A Contratar ISE 10	PL; TP	TP1; PL1	30TP; 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	20	55	0	0	0	0	0	182

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

- Basic OOP concepts (inheritance, encapsulation, object state, object behavior, polymorphism, etc.)
- Read and draw UML diagrams and classes and objects

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

The main objective of this curricular unit of the Market Technologies curricular unit is to cement the knowledge of OOP, taking full advantage of the specificities offered by the Java execution virtual environments.

At the end of the course, students should be able to:

- ¿ Identify the different primitive types provided by the Standard Edition of Java;
- ¿ Apply good programming practices;
- ¿ Implement different flow controls: Branching and Loops;
- ¿ Consult the definitions of the different types offered by Standard Edition using Java API documentation;
- ¿ Define your own Java types;
- ¿ Handle and define exceptions;
- ¿ Be able to develop Object Oriented solutions, drawing UML class diagrams, implementing Java classes taking advantage of concepts such as inheritance, polymorphism, abstract contracts and generic types.

Syllabus

Introduction

- Introduction to the history of Java
- Java code vs byte codes vs Java Virtual Machine

basic computing

- different Java types
- operations on character strings
- conventions
- helpful comments

Flow Control:

- branching
- primitive types and reference
- boolean primitive type
- use of simple enumerated
- implement loops (while, do, for)

Definition of classes and methods

- create objects, classes and methods
- UML class diagrams
- builders
- static methods and static variable definition

Arrays

- Sorting and searching arrays
- Multidimensional arrays

Inheritance, Polymorphism and Interfaces

- concepts of polymorphism and inheritance
- Definition of interfaces
- Dynamic Binding in Java
- Derived classes

Exceptions

- exception handling
- Errors vs Exceptions vs Runtime Exceptions ; Checked and Unchecked exceptions

Dynamic and generic data structures

- ArrayList instances
- Single or double linked data structures

- Iterators

Teaching methodologies (including evaluation)

Within the scope of the UC, the following methodologies will be used:

- ¿ Theoretical-practical presentation of the subject;
- ¿ Discussion of the topics covered and clarification of doubts;
- ¿ Encouragement of participation, interaction, and group dynamics;
- ¿ Carrying out practical work to apply knowledge;

The final classification will be computed as follows:

- ¿ Final Exam ¿ minimum of 9.5 values (40%);
- ¿ Practical component ¿ evaluated by an exercise sheet and a final assignment (60%);

The grade will be awarded after discussion with the teacher about all the practical work carried out throughout the course;

Main Bibliography

Walter Savitch, Java: An introduction to problem solving & Programming. Pearson; 8ª edição