

Caracterização da Unidade Curricular / Characterization of the Curricular Unit

Designação da Unidade Curricular (UC) / Title of Curricular Unit (CU): Linguagens de Programação / Programming Languages

Área científica da UC / CU Scientific Area: Ciências Informáticas / Computer Science

Semestre / Semester: 2º

Número de créditos ECTS / Number of ECTS credits: 6

Carga horária por tipologia de horas / Workload by type of hours: TP: 45; OT: 6; O: 9

Carga letiva semanal / Weekly letive charge: 3h

Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):

Os objetivos específicos pretendidos com a UC de Linguagem de Programação é que ao final da UC os alunos consigam propor, implementar e/ou aperfeiçoar programas/sistemas informáticos, através da utilização de aplicações, ambientes de desenvolvimento integrado e técnicas de programação, aplicando na prática os conhecimentos obtidos na UC, como: resolução de algoritmos simples e complexos, programação estruturada e programação orientada à objetos. Para tal, os alunos desenvolvem durante a UC, diversas aplicações voltadas para plataformas locais e em ambiente Web, simulando a resolução de problemas lógicos em diversos contextos, de modo a colocar em prática os conhecimentos de programação adquiridos.

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

The specific objectives with the UC of Programming Language are that at the end of the UC students can propose, implement and/or improve software/computer systems, through the use of applications, integrated development environments and programming techniques, applying in practice the knowledge obtained in the UC, such as: resolution of simple and complex algorithms, structured programming and object-oriented programming. To this end, students develop during the UC, several applications aimed at local platforms and in the Web environment, simulating the resolution of logical problems in various contexts, in order to put into practice the knowledge of acquired programming.

Conteúdos programáticos:

1. Introdução à programação orientada à objetos utilizando Java:

- 1.1 Tipos de dados
- 1.2 Operadores
- 1.3 Parâmetros
- 1.4 Estruturas de repetição
- 1.5 Estruturas de dados
- 1.6 Vetores
- 1.7 Matriz
- 1.8 Listas encadeadas

1.9 Exercícios práticos envolvendo todos os conteúdos

2. IDE para desenvolvimento em Java:

2.1 Eclipse

2.2 NetBeans

2.3 Desenvolvimento de Interfaces gráficas com Java Swing e JavaFX

3. Orientação a Objetos usando Java:

3.1 Introdução

3.2 Abstração

3.3 Definições de Objetos

3.4 Classes

3.5 Atributos

3.6 Utilização de Objetos e Classes

3.7 Escopo de Classes e Escopo de Instância

3.8 Encapsulamento

3.9 Herança

3.10 Polimorfismo

3.11 Threads

3.12 Tratamentos de exceções

3.13 Exercícios práticos envolvendo todos os conteúdos

4. Desenvolvimento Web com Java Web:

4.1 Java EE e GlassFish Server

4.2 Desenvolvimento de Servlets em Java Web

4.3 Métodos GET e POST em Java Web

4.4 Desenvolvimento de Web com páginas JSP

Syllabus:

1. Introduction to object-oriented programming using Java:

1.1 data types

1.2 operators

1.3 Parameters

1.4 repeat structures

1.5 data structures

1.6 Vectors

1.7 Matrix

1.8 linked lists

1.9 Practical exercises involving all the contents

2. Java Development IDE:

- 2.1 Eclipse
- 2.2 NetBeans
- 2.3 Graphical Interface Development with Java Swing and JavaFX

3. Object Orientation using Java:

- 3.1 Introduction
- 3.2 withdrawal
- 3.3 Object Definitions
- 3.4 classes
- 3.5 Attributes
- 3.6 Use of objects and classes
- 3.7 Class Scope and Dispute Scope
- 3.8 Encapsulation
- 3.9 Heritage
- 3.10 Polymorphism
- 3.11 Threads
- 3.12 treatment of exceptions
- 3.13 Practical exercises involving all the contents

4. Developer Web with Java Web:

- 4.1 Java EE and GlassFish Server
- 4.2 Development of Java Web Servlets
- 4.3 GET and POST methods in Java Web
- 4.4 Web development with JSP pages