



UNIVERSIDADE LUSÍADA DE LISBOA

Programa de Unidade Curricular

- Ano Lectivo 2007/2008 -

Faculdade

Ciências da Economia e da Empresa

Licenciatura

Informática

Unidade Curricular

Programação Avançada

Ano: 2.º

Tipo: 2.º Semestre

Nº ECTS: 6

Regente

Prof. Doutor Paulo Enes da Silveira

Assistente

-

Carga Horária Lectiva Semanal

Aulas Teóricas: -

Aulas Teórico-práticas: 2

Orientação Tutorial: 1

Língua de Ensino

Português

Objectivos Gerais

1. Compreender a complexidade dos problemas a resolver e as soluções de software necessárias no início do séc. XXI.
2. Estudar os paradigmas da programação mais recentes e os desenvolvimentos actuais.
3. Praticar programação com a linguagem C# e Java.

Objectivos Específicos

1. Reconhecer as crises de software na programação e as suas soluções.
2. Compreender aspectos avançados da Programação Orientada por Objectos.
3. Abordar a Metaprogramação e a Programação por Contractos.
4. Estudar a Programação por Padrões.
5. Estudar a Programação por Aspectos e sua aplicação.
6. Prática de alguns dos paradigmas em C# e Java.



UNIVERSIDADE LUSÍADA DE LISBOA

Competências a adquirir

1. Reconhecimento das dificuldades de Software para a resolução dos complexos problemas da actualidade e possíveis soluções.
2. Melhor conhecimento do paradigma Orientado por Objectos e conhecimentos da Metaprogramação, da Programação por Contractos, da Programação por Padrões e da Programação por Aspectos.
3. Melhor conhecimento das linguagens C# e Java, com implementação prática de programas.

Metodologia de Ensino

1. As aulas teóricas-práticas abordam a exposição e estudo das matérias acompanhadas com exemplos práticos, apontamentos e referências bibliográficas.
2. As aulas tutoriais são dedicadas aos estudos de casos de aplicação dos paradigmas com implementação prática nas linguagens C# e Java.

Programa da Unidade Curricular / Conteúdo programático

1. Estudo de novas abordagens ou paradigmas da programação
 - 1.1. Estudo da evolução da Programação e seus desafios para o séc.XXI : Discussão sobre as crises de software e os desenvolvimentos necessários para o futuro.
2. Programação Orientada por Objectos – POO: aspectos avançados e sua prática em C#.
3. Referências à Metaprogramação e Programação por Contratos.
4. Programação por Padrões (Patterns).
 - 4.1 Introdução ao conceito de Padrão.
 - 4.2 Classificação de Padrões.
 - 4.2.1. "Fundamental Patterns".
 - 4.2.2. "Creational Patterns".
 - 4.2.3. "Structural Patterns".
 - 4.2.4. "Behavioral Patterns".
 - 4.2.5. "Architectural Patterns".
 - 4.2.6. Estudo de casos com Padrões em Java e C#.
5. Programação Orientada por Aspectos – POA.
AOP - Aspect Oriented Programming).
 - 5.1. Introdução à POA.
 - 5.2. Noção de Aspecto.
 - 5.3. Exemplos de "Cross-Cutting".
 - 5.4. Definição de conceitos da POA:
 - 5.4.1. Responsabilidades (concerns).
 - 5.4.2. Separação de responsabilidades (separation of concerns).
 - 5.4.3. Responsabilidades transversais (crosscutting concerns).
 - 5.5. As três fases distintas da POA:
 - 5.5.1. Decomposição.
 - 5.5.2. Implementação.
 - 5.5.3. Recomposição.
 - 5.6. Introdução ao AspectJ.



UNIVERSIDADE LUSÍADA DE LISBOA

- 5.6.1. Join points.
- 5.6.2. Pointcuts.
- 5.6.3. Advices.
- 5.6.4. Aspects.
- 5.7. POA em ambiente .NET
 - 5.7.1. Introdução
 - 5.7.2. O compilador EOS, extensão para C#
- 5.8. Estudo de casos com POA

Programação com a linguagem C# e Java, ao longo da disciplina.

Bibliografia Principal

Autor(es)

Gamma, E., Helm, R., Johnson, R., Vlissides, J.,

Título

Design Patterns: Elements of Reusable Object-Oriented Software

Edição

Local

Editora

Addison Wesley

Ano

1995

Autor(es)

Filman, RE., Elrad. T., Clarke, S., Aksit, M.

Título

Aspect-Oriented Software Development

Edição

1.^a

Local

Boston

Editora

Addison Wesley



UNIVERSIDADE LUSÍADA DE LISBOA

Ano

2005

Bibliografia Complementar

Autor(es)

Marques, P., Pedroso, H., C# 2.0

Título

C# 2.0

Edição

1.^a

Local

Editora

Addison Wesley

Ano

2005

Metodologia de Avaliação Contínua / Elementos relevantes

Avaliação Contínua (AC) vale 60% da nota final (NF) se $AC \geq 8$ e $AC > \text{Frequência}(F)$

Testes escritos: 30%;

Trabalhos teóricos e práticos (TT e TP): 25%;

Oral: 25%;

Conferências, visitas: 10%;

Outros (componente prática): 10%.

As componentes 1 e 2 são obrigatórias.

A presença nas aulas deve ser ≥ 70 para validar a AC. A F vale 40% da NF com a AC validada. Se a AC não for válida, a F vale 70% e o TP vale 30% da NF. A não aprovação na AC e F leva a Exame, sendo Exame 70% e TP 30% da NF.

Recursos Didáticos

Computadores nas aulas práticas com os compiladores C (C++ do VisualStudio) C# e Java;
Dispositivo projector de conteúdos digitais (Data-show), ligado ao computador.

Palavras-chave

Programação Avançada, Programação Orientada por Objectos; Programação Orientada por Aspectos; Programação por Padrões