

Informação Geral:

| | | | |
|-------------------------|--|-------------------|--------------|
| Unidade Orgânica | Escola Superior de Tecnologia e Gestão | Ano Letivo | 2015/2016 |
| Curso | Licenciatura em Engenharia Informática | Grau | Licenciatura |
| UC/Módulo | Algoritmos e Estruturas de Dados | ECTS | 6 |
| Área Científica | Sistemas de Informação | Carácter | Obrigatório |
| Horas Totais | 162 | T 0 | TP 30 |
| | | PL 45 | TC 0 |
| | | S 0 | E 0 |
| | | OT 0 | |

T - Ensino Teórico; TP - Teórico Prático; PL - Prático e Laboratorial; TC - Trabalho de Campo; S - Seminário; E - Estágio; OT - Orientação Tutorial

Docente responsável:

José Manuel Magno Lopes

Docentes que lecionam a unidade curricular:

José Manuel Magno Lopes (120,00 horas semanais de contacto: TP: 6,00; TP: 24,00; PL: 45,00; PL: 45,00;)

António Carlos Alves Urbano (105,00 horas semanais de contacto: TP: 24,00; PL: 45,00; PL: 36,00;)

Marco Paulo Monteiro Ferreira (135,00 horas semanais de contacto: PL: 45,00; PL: 45,00; PL: 45,00;)

Pré-requisitos:

Não tem.

Idioma:

Português e Inglês

Enquadramento:

Esta UC pertence ao 2º ano e complementa os conceitos apreendidos no 1º ano em Programação I e Programação II. O estudante adquire competências para utilização e implementação de algoritmos e estruturas de dados na resolução dos problemas de engenharia de software.

Objetivos de aprendizagem:

- C1. Conhecimentos sobre Programação Orientada aos Objetos
- C2. Capacidade de desenvolver aplicações utilizando arquiteturas de software (MVC)
- C3. Capacidade de armazenar (gravar e ler) eficientemente dados através de streams
- C4. Estudo de conceitos básicos e fundamentos sobre estruturas de dados e algoritmos
- C5. Capacidade de analisar estratégias de implementação destas estruturas de dados
- C6. Capacidade de aplicar as estruturas de dados e algoritmos mais apropriados no desenvolvimento de projetos
- C7. Capacidade de analisar a complexidade de um algoritmo

Programa:

Conteúdos Programáticos:

1. Breve revisão sobre POO
2. Arquitetura de software MVC
3. Noções básicas de Streams
4. Estruturas de dados genéricas
 - 4.1. Tabelas (revisões)
 - 4.2. Listas
 - 4.3. Pilhas e filas
 - 4.4. Tabelas de hash
 - 4.5. Árvores binárias
5. Análise de algoritmos
 - 5.1. Comparar diferentes algoritmos que resolvem o mesmo problema
 - 5.2. Notação Big-Oh (simplificada)
6. Recursividade
 - 6.1. Conceito de recursividade
 - 6.2. Vantagens e desvantagens da sua utilização

Fundamentação da coerência dos conteúdos programáticos com os objetivos/competências da unidade curricular:

Os conteúdos programáticos lecionados contribuem para as competências gerais estabelecidas para a UC da seguinte forma:

1. Breve revisão sobre POO (C1)
2. Arquitetura de software MVC (C2)
3. Noções básicas de Streams (C3)
4. Estruturas de dados genéricas (C4, C5, C6)
 - 4.1. Tabelas (revisões)
 - 4.2. Listas
 - 4.3. Pilhas e filas
 - 4.4. Tabelas de hash
 - 4.5. Árvores binárias
5. Análise de algoritmos
 - 5.1. Comparar diferentes algoritmos que resolvem o mesmo problema – (C7, C8)
 - 5.2. Notação Big-Oh (simplificada) – (C7, C8)
6. Recursividade
 - 6.1. Conceito de recursividade – (C5, C7, C8)
 - 6.2. Vantagens e desvantagens da sua utilização – (C5, C7, C8)

Metodologia de Ensino / Aprendizagem:

Presencial:

1. Ensino teórico
 - 1.1. Apresentação dos conceitos sobre algoritmos e estruturas de dados
 - 1.2. Exemplificação e aplicação a problemas reais
2. Ensino prático
 - 2.1. Resolução de problemas através da programação de algoritmos e aplicação de estruturas de dados
 - 2.2. Análise crítica dos resultados

3.Orientação tutorial

3.1.Sessões de orientação pessoal para conduzir o processo de aprendizagem e esclarecerem-se dúvidas

Autónoma:

1.Estudo

1.1.Leitura da bibliografia indicada e resolução de exercícios de forma a aprofundar e complementar o conhecimento

1.2.Resolução dos exercícios recomendados pela unidade curricular

2.E-aprendizagem

2.1.Consulta de material relativo à unidade curricular

Recursos Específicos:

1.Ensino teórico – sala de aula normal

2.Ensino prático – laboratório informático com computadores contendo:

2.1.JDK (Java Development Kit)

2.2.Android Studio

3.Orientação tutorial – gabinete ou sala de aula normal

Avaliação:

Descrição:

Avaliação Periódica:

- componente teórica (30% - mínimo 10)

- 2 provas escritas teóricas (melhor nota das duas - 30%)

- componente prática (70%)

- 4 fichas práticas (17,5% cada)

Avaliação Final:

- 1 prova escrita teórica (30% - mínimo 10)

- 1 prova prática em computador (70%)

Os estudantes que assim pretendam, estão dispensados da realização dos elementos de avaliação incluídos no método de avaliação da unidade curricular, desde que estes tenham sido realizados no próprio ano letivo, e que os estudantes tenham obtido os mínimos.

As melhorias de classificação devem ser realizadas integralmente, não sendo salvaguardadas notas parcelares para este efeito.

| | |
|--|---|
| Número de elementos de avaliação final: | 2 |
|--|---|

| | |
|---|---|
| Número de elementos de avaliação contínua/periódica: | 6 |
|---|---|

Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular:

As metodologias de ensino usadas contribuem para as competências gerais estabelecidas para a UC da seguinte forma:

Presencial

1.Ensino teórico

1.1.Apresentação dos conceitos sobre algoritmos e estruturas de dados – (C1, C4, C5, C7)

1.2.Exemplificação e aplicação a problemas reais – (C6, C7)

2.Ensino prático

2.1.Resolução de problemas através da programação de algoritmos e aplicação de estruturas de dados – (C2, C3, C5, C8)

2.2.Análise crítica dos resultados – (C6, C7)

2.3.As últimas aulas serão de apoio ao projeto – (C6, C8)

3.Orientação tutorial

3.1.Sessões de orientação pessoal para conduzir o processo de aprendizagem e esclarecerem-se dúvidas – (C1, C2, C3, C4,

C5, C6, C7, C8)

Autónoma

1. Estudo

1.1. Leitura da bibliografia indicada e resolução de exercícios de forma a aprofundar e complementar o conhecimento – (C4, C5, C6, C7)

1.2. Resolução dos exercícios recomendados pela unidade curricular – (C6, C8)

2. E-aprendizagem

2.1. Consulta de material relativo à unidade curricular – (C4, C5, C6, C7)

3. Reunião com o colega de grupo de forma a dividirem tarefas do projeto e estudarem em conjunto os diversos conceitos – (C2, C3, C5, C6, C7, C8)

Bibliografia:

Recomendada:

- Data structures & problem solving using Java (4ª edição), Mark Allen Weiss, Addison Wesley, 2009, ISBN: 978-0321541406
- Data Structures and Algorithms in Java (3ª edição), Adam Drozdek, Cengage Learning Asia, 2008, ISBN: 978-9814239233
- Thinking in Java (4ª edição), Bruce Heckel, Prentice-Hall, 2006, ISBN: 978-0131872486
- Data structures & algorithm analysis in Java, Mark Allen Weiss, Addison Wesley, 1998, ISBN: 978-0201357547
- Data structures in Java, Thomas Standish, Addison Wesley, 1997, ISBN: 978-0201305647
- Algorithms in C (3ª edição), Robert Sedgewick, Addison Wesley, 1997, ISBN: 978-0201314526

Complementar:

- Introduction to algorithms (3ª edição), Cormen, Leiserson and Rivest, MIT Press, 2009, ISBN: 978-0262033848
- Programação Orientada aos Objectos em Java 2, F. Mário Martins, FCA, 2000, ISBN: 978-9727221967
- The Java Tutorials, <http://download.oracle.com/javase/tutorial/>
- Understanding Object-Oriented Programming with Java, Timothy Budd, Addison-Wesley, 1999, ISBN: 978-0201612738
- Introduction to Programming Using Java: An Object-Oriented Approach (2ª edição), David M. Arnow, Geral Weiss, Addison-Wesley, 2003, ISBN: 978-0321200068
- The Java Tutorial (2ª edição), Mary Campione & Kathy Wallrath, Sunsoft Press, 1998, ISBN: 978-0201310078
- The Java Handbook, Patrick Naughton, Osborne/Mc Graw-Hill, 1996, ISBN: 978-0078821998