

<p>POLI ESCOLA SUPERIOR TECNOLOGIA GESTÃO TÉCNICO GUARDA</p>	<p>GUIA DE FUNCIONAMENTO DA UNIDADE CURRICULAR (GFUC)</p>	<p>MODELO PED.008.03</p>
--	--	--

<i>Curso</i>	Engenharia Informática						
<i>Unidade curricular</i> (UC)	Algoritmos e Estruturas de Dados						
<i>Ano letivo</i>	2023/2024	<i>Ano</i>	1.º	<i>Período</i>	1.º semestre	<i>ECTS</i>	6
<i>Regime</i>	Obrigatório	<i>Tempo de trabalho (horas)</i>		Total: 168	<i>Contacto: 75</i>		
<i>Docente(s)</i>	Doutor Paulo Jorge Costa Nunes						
<input type="checkbox"/> <i>Responsável da UC ou</i>	<input checked="" type="checkbox"/> <i>Coordenador(a) Área/Grupo Disciplinar</i>		Doutor José Carlos Coelho Martins da Fonseca (Programação e Multimédia)				
<input type="checkbox"/> <i>Regente (cf. situação de cada Escola)</i>							

GFUC PREVISTO

1. OBJETIVOS DE APRENDIZAGEM

1. Escrever algoritmos utilizando pseudocódigos e simbologia de fluxogramas
2. Diferenciar estruturas de dados e escolher a melhor para cada algoritmo em concreto
3. Analisar a complexidade de algoritmos
4. Explicar e aplicar algoritmos de ordenação e pesquisa
5. Implementar algoritmos numa linguagem de programação

J. C. Martins

2. CONTEÚDOS PROGRAMÁTICOS

1. Introdução ao desenvolvimento de algoritmos
 1. Conceito de algoritmo
 2. Fases de desenvolvimento de um algoritmo
2. Linguagem algorítmica
 1. Importância, sintaxe
 2. Fluxograma e pseudolinguagem
 3. Tipos de variáveis simples, tipos de variáveis estruturados
 4. Instruções de entrada e saída
 5. Estruturas de controlo
3. Fluxogramas
 1. Simbologia
 2. Aplicações
4. Pseudolinguagem
 1. Introdução
 2. Pseudocódigos e exemplos
5. Estruturas de dados
 1. Cadeias de caracteres
 2. Registos e Ficheiros
 3. Vetores
 4. Matrizes ou tabelas
 5. Listas ligadas
 6. Filas
 7. Pilhas
 8. Árvores

6. Análise de complexidade de algoritmos
 1. Complexidade espacial e temporal
 2. Complexidade e eficiência
 3. Notações “*Big O*” e “*Little o*”
 4. Crescimento de funções
7. Ordenação, seleção e pesquisa
 1. Pesquisa sequencial e binária
 2. Algoritmos de ordenação
 3. Conceitos sobre algoritmos de ordenação
 4. Análise de complexidade de alguns algoritmos de ordenação
 5. Comparação dos algoritmos de ordenação

J. Nunes

3. DEMONSTRAÇÃO DA COERÊNCIA DOS CONTEÚDOS PROGRAMÁTICOS COM OS OBJETIVOS DA UC

Os conteúdos 1 a 4 são coerentes com o objetivo 1, porque é definido o conceito de algoritmo, são apresentadas as fases necessárias para desenvolver algoritmos, são descritos os elementos da linguagem algorítmica e são apresentadas duas formas de escrever algoritmos, utilizando pseudocódigos e fluxograma.

O conteúdo 5 é coerente com o objetivo 2, “Diferenciar estruturas de dados e escolher a melhor para cada algoritmo em concreto”, porque são apresentadas as estruturas de dados fundamentais utilizadas nos algoritmos.

Os conteúdos 6 e 7, são coerentes com o objetivo 3, porque são apresentados os conceitos sobre complexidade e eficiência assintótica de algoritmos, notações de análise “*Big O*” e “*Little o*” e descrita uma análise comparativa de complexidade de algoritmos de ordenação.

O conteúdo 7 é coerente com o objetivo 4, porque no referido capítulo são descritos e comparados algoritmos de ordenação e apresentados métodos de pesquisa.

Os conteúdos 2, 5 e 7, são coerentes com o objetivo 5, porque são apresentados os conceitos básicos e estruturas necessárias de uma linguagem de programação para implementar algoritmos simples.

4. BIBLIOGRAFIA PRINCIPAL

Obrigatória

- [1] Nunes Paulo (2011) “Manual de Algoritmos e Estruturas de Dados – Engenharia Informática”. Guarda, IPG.
- [2] Vasconcelos, J.B. e Carvalho, J. V. (2005). Algoritmia e estrutura de Dados – Programação nas Linguagens C e JAVA. Lisboa: Centro Atlântico.

<p>POLI ESCOLA SUPERIOR TECNOLOGIA GESTÃO</p> <p>TÉCNICO GUARDA</p>	<p>GUIA DE FUNCIONAMENTO DA UNIDADE CURRICULAR (GFUC)</p>	<p>MODELO PED.008.03</p>
--	--	--

- [3] Manzano, J.A. e Oliveira, J.F. (2005). Algoritmos – Lógica para desenvolvimento de Programação de Computadores. 17.ª Edição. São Paulo: Érica.
- [4] Jeffrey Elkner, Allen B. Downey, and Chris Meyers (2012). "How to Think Like a Computer Scientist: Learning with Python". 2nd Edition documentation. <http://www.openbookproject.net/thinkcs/python/english2e/>
- [5] Al Sweigart. Invent Your Own Computer Games with Python, 2nd Edition by <http://inventwithpython.com/downloads/>.

Recomendada

- [1] Magri, J.A. (2003). Lógica de Programação – Ensino Prático. São Paulo: Érica.
- [2] Lopes, A. E Garcia, G. (2002). Introdução à Programação – 500 Algoritmos Resolvidos. 5ª Edição. Rio de Janeiro: Elsevier.
- [3] Knuth, Donald E. (1998). "The Art of Computer Programming – VOLUME 1- Fundamentals Algorithms". Third Edition, ADDISON – WESLEY.
- [4] Knuth, Donald E. (1993). The Art of Computer Programming – VOLUME 3 - Sorting and Searching. Third Edition, Prentice Hall.
- [5] Goodman, S.E. and Hedetniemi, S.T.(1993). Introduction to the Design and Analysis of Algorithms. Prentice Hall.
- [6] Paul Gries, Jennifer Campbell, Jason Montojo, Practical Programming (2nd edition) An Introduction to Computer Science Using Python 3 <https://pragprog.com/book/gwpy2/practical-programming>
- [7] John M. Zelle, Python Programming: An Introduction to Computer Science. <http://mcsp.wartburg.edu/zelle/python/>

5. METODOLOGIAS DE ENSINO (REGRAS DE AVALIAÇÃO)

Lição expositiva, lição interativa, resolução de problemas, trabalho individual e trabalho de grupo.

REGRAS DE AVALIAÇÃO

1. Contínua, exame época normal
 - a) Frequência (50%) – Marcada pela Direção da ESTG.
 - b) Teste online 1 no Moodle (10%) (6/11/2023)
 - c) Teste online 2 no Moodle (10%) (4/12/2023)
 - d) Trabalho de grupo (30%, Formato: zip+pdf, Apresentação: 08-01-2024)
 - Resolução de um problema.
 - Máximo dois alunos;
 - Programa;
 - Apresentação (PowerPoint) e defesa.
2. Exame época recurso - 100%.

6. DEMONSTRAÇÃO DA COERÊNCIA DAS METODOLOGIAS DE ENSINO COM OS OBJETIVOS DA UNIDADE CURRICULAR

<p>POLI ESCOLA SUPERIOR TECNOLOGIA GESTÃO TÉCNICO GUARDA</p>	<p>GUIA DE FUNCIONAMENTO DA UNIDADE CURRICULAR (GFUC)</p>	<p>MODELO PED.008.03</p>
--	--	--

Lição expositiva está de acordo com os objetivos porque é necessário apresentar os conhecimentos teóricos da unidade curricular aos alunos.

Lição interativa está coerente com os objetivos porque a interação dos alunos com o docente facilita a aprendizagem dos conceitos. São apresentados algoritmos focando cada um dos aspetos das suas fases de escrita, as estruturas de dados utilizadas, os métodos de pesquisa e ordenação e análise de complexidade. Os alunos podem questionar o docente sobre qualquer elemento do algoritmo, trocar e enriquecer ideias permitindo aumentar os conhecimentos que cada um tem.

Resolução de problemas está coerente com os objetivos porque permite ao aluno a aplicação dos conhecimentos teóricos na escrita de algoritmos que envolve a escolha das estruturas de dados mais adequadas, análise de complexidade e em alguns casos a aplicação de algoritmos de ordenação e pesquisa. A resolução de problemas é apoiada por uma aplicação web disponibilizada pelo docente que facilita a escrita de algoritmos em linguagem algorítmica devidamente documentados.

Trabalho individual está coerente com os objetivos porque a realização de um trabalho individual que consiste no desenvolvimento de um algoritmo, documentação de todas as suas fases de desenvolvimento, elaboração de uma apresentação e defesa permitem ao aluno solidificar os seus conhecimentos adquiridos na unidade curricular e desenvolver a sua capacidade individual na resolução de problemas em geral através da escrita de algoritmos.

Trabalho de grupo está coerente com os objetivos porque permite ao aluno desenvolver sua capacidade de trabalhar em grupo e reconhecer as suas vantagens. Na realização do trabalho os alunos têm de resolver um problema com alguma dimensão onde é necessário desenvolver diversos algoritmos com todas as suas fases. Na realização deste trabalho os alunos são obrigados a aplicar a maioria dos conhecimentos adquiridos.

Os alunos são incentivados a escolher temas para os trabalhos de outras unidades curriculares (Álgebra e Geometria Analítica, Introdução à Física, Introdução à Programação), podendo escolher qualquer outro tema. A realização dos trabalhos de grupo e individuais é acompanhada pelo docente em orientações tutoriais. A apresentação dos trabalhos realizados pelos alunos é realizada em sala de aula com as presenças dos seus colegas. Isto permite ao aluno ver a aplicação dos conhecimentos teóricos e alargar a sua capacidade na resolução de problemas semelhantes.

7. CONTATOS E HORÁRIO DE ATENDIMENTO

Nome	Email	Telefone	Gabinete	Horário de atendimento
Paulo Nunes	pnunes@ipg.pt		20	Segunda-feira: 8h30-10h00; 17h30-18h30 Terça-feira: 17h30-18h00 Quinta-feira: 17h00-18h00

Data: 15/09/2023

Docentes da disciplina

Coordenador da área disciplinar

Paulo Jorge Costa Nunes

Professor Adjunto Paulo Jorge Costa Nunes

Professor Coordenador José Carlos Coelho Martins da Fonseca

**GUIA DE FUNCIONAMENTO
DA UNIDADE CURRICULAR**
(GFUC)

MODELO
PED.008.03

(Programação e Multimédia)