
Ano Letivo 2021-22

Unidade Curricular ALGORITMIA E ESTRUTURAS DE DADOS

Cursos TECNOLOGIAS INFORMÁTICAS

Unidade Orgânica Instituto Superior de Engenharia

Código da Unidade Curricular 19321007

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 (PT)

Modalidade de ensino

Obrigatória

Docente Responsável

Roberto Célio Lau Lam

DOCENTE	TIPO DE AULA	TURMAS	TOTAL HORAS DE CONTACTO (*)
Roberto Célio Lau Lam	PL; TP	TP1; PL1	14TP; 42PL

* 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	S2	14TP; 42PL	156	6

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

Precedências

Sem precedências

Conhecimentos Prévios recomendados

Conhecimentos de Programação. Raciocínio lógico e dedutivo.

Objetivos de aprendizagem (conhecimentos, aptidões e competências)

Fornecer aos alunos:

Conhecimentos sobre metodologia de programação orientada por objetos.

Capacidade de construir programas nesta metodologia de programação. Saber conceber, analisar, e realizar, algoritmos lógicos para resolução de problemas de pequena, média dimensão.

Conhecimentos e saber utilizar a anotação descritiva da eficiência algorítmica.

Conhecimentos e saber utilizar os algoritmos básicos de ordenação e pesquisa: Bubble sort, Shell sort, Quick sort e as pesquisas sequencial e binária.

Conhecimentos e saber utilizar estruturas de armazenamento lineares: vetores, pilhas, filas e hierárquicas: árvores.

Conhecimento sobre estruturas de armazenamento híbridas nomeadamente: Tabelas de dispersão (Hash Tables), grafos, matrizes esparsas

Conhecer situações de utilização prática das estruturas de armazenamento híbridas: determinação do caminho mais curto (matrizes e listas de adjacência).

Conteúdos programáticos

1. Programação Orientada por Objetos. Formalização teórica, exemplificação com JAVA: Classes, objetos. Variáveis, funções membro (métodos, interfaces), construtores e destruidor, sobrecarga de funções membro. Atributos de acesso. Operadores, sobrecarga de operadores, Herança, Polimorfismo.
2. JAVA (utilização memória dinâmica, ficheiros e interfaces gráficas do utilizador - GUI).
3. Complexidade algorítmica.
4. Algoritmos de ordenação em vetores: Bubble sort, Shell sort e Quick sort
5. Algoritmos de pesquisa em vetores
6. Listas, sob os conceitos LIFO, FIFO (Filas e Pilhas).
7. Árvores. Terminologia. BST árvores de pesquisa binária. Percursos: prefixo, infixo e pósfixo. Implementação. Árvores AVL
8. Tabelas dispersão e grafos. Terminologia e exemplos de implementação.
9. Utilização prática do conteúdo da disciplina em aplicações práticas: Caminho mais curto. Estruturas para armazenamento de: listas de adjacência e matrizes de adjacências.

Metodologias de ensino (avaliação incluída)

Aulas Teórico-Práticas: Método expositivo, orientado de acordo com um ensino baseado em problemas, com projeção e explicação dos objetivos e conteúdos correspondentes a cada semana.

Aulas Práticas: Prática laboratorial em computador. Resolução e codificação de exercícios/problemas práticos tipificados, selecionados em conformidade com o conteúdo teórico semanal.

Avaliação

Componentes da avaliação classificadas de 0-20 valores:

- Trabalho(s) prático(s) (TP)
- Prova escrita (PE): Teste / Exame

Classificação mínima em cada uma das componentes 7 valores.

Nota final: 0.5 PE + 0.5 TP

Bibliografia principal

Cormen, T.H., Leiserson, C.E., Rivest, R. and Stein, C., Introduction to Algorithms, MIT Press.

Sedgewick, R., Addison-Wesley Publishing Company, Inc.

Eckels, B., Thinking in JAVA I e II, 3th Edition.

Estruturas de Dados e Algoritmos em JAVA A. Adrego da Rocha, 2011, FCA.

Rodrigues P., Pereira P. E Sousa M., Programação em C++ Conceitos básicos e Algoritmos. FCA.

Guerreiro, P., Elementos de Programação com C, FCA Lidel. 3ª 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 (PT)

Teaching/Learning modality Required

Coordinating teacher Roberto Célio Lau Lam

Teaching staff	Type	Classes	Hours (*)
Roberto Célio Lau Lam	PL; TP	TP1; PL1	14TP; 42PL

* For classes taught jointly, it is only accounted the workload of one.

Contact hours	T	TP	PL	TC	S	E	OT	O	Total
	0	14	42	0	0	0	0	0	156

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

Programming knowledge. Logical and deductive reasoning.

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

Provide students with:

Knowledge of object-oriented programming methodology.

Ability to build programs in this programming methodology. Know how to design, analyze and implement logical algorithms for solving small and medium-sized problems.

Knowledge and knowing how to use descriptive annotation of algorithmic efficiency.

Knowledge and know how to use the basic sorting and search algorithms: Bubble sort, Shell sort, Quick sort and sequential and binary searches.

Knowledge and knowing how to use linear storage structures: vectors, stacks, queues and hierarchical: trees.

Knowledge of hybrid storage structures, namely: Hash Tables, graphs, sparse matrices.

Know situations of practical use of hybrid storage structures: determination of the shortest path, multiplication of sparse matrices. At the end of the course, they should be able to scale efficient small and medium data storage structures.

Syllabus

1. Object Oriented Programming. Theoretical formalization, exemplifying with JAVA: Classes, objects. Variables, member functions (methods, interfaces), constructors and destructor, member function overloading. Access attributes. Operators, Operator overloading, Inheritance, Polymorphism.
 2. JAVA (use of dynamic memory, files, and graphical user interfaces - GUI).
 3. Algorithmic complexity.
 4. Vector sorting algorithms: Bubble sort, Shell sort and Quick sort
 5. Vector Search Algorithms
 6. Lists, under the concepts LILO, FIFO (Queues and Stacks).
 7. Trees. Terminology. BST binary search trees. Routes: prefix, infix and postfix. Implementation. AVL trees
 8. Hash tables and graphs. Terminology and implementation examples.
 9. Practical use of course content in practical applications: shortest path problem, structures for storing: adjacency lists and sparse adjacency matrices.
-

Teaching methodologies (including evaluation)

Theoretical-Practical Classes: Lecture method, guided according to problem-based teaching, with projection and explanation of the objectives and corresponding content each week.

Practical classes: Computer laboratory practice. Resolution and coding of typified practical exercises/problems, selected in accordance with the weekly theoretical content.

Evaluation

Assessment components rated from 0-20 points:

- Practical assignment(s) (TP)
- Written test (PE): Test / Exam

Minimum classification in each of the components 7 points.

Final grade: 0.5 PE + 0.5 TP

Main Bibliography

Cormen, T.H., Leiserson, C.E., Rivest, R. and Stein, C., Introduction to Algorithms, MIT Press.

Sedgewick, R., Addison-Wesley Publishing Company, Inc.

Eckels, B., Thinking in JAVA I e II, 3th Edition.

Estruturas de Dados e Algoritmos em JAVA A. Adrego da Rocha, 2011, FCA.

Rodrigues P., Pereira P. E Sousa M., Programação em C++ Conceitos básicos e Algoritmos. FCA.

Guerreiro, P., Elementos de Programação com C, FCA Lidel. 3ª Edição.