

Unidade Curricular	Algoritmos e Estruturas de Dados		Área Científica	Ciências Informáticas	
CTeSP em	Desenvolvimento de Software		Escola	Escola Superior de Tecnologia e Gestão de Bragança	
Ano Letivo	2019/2020	Ano Curricular	1	Nível	0-1
Tipo	Semestral	Semestre	2	Créditos ECTS	3.0
		Código	4088-640-1202-00-19		
Horas totais de trabalho	81	Horas de Contacto	T -	TP 10	PL 20
			TC -	S -	E -
			OT -	O -	

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

Nome(s) do(s) docente(s) Paulo Duarte Ferreira Gouveia, Evandro Pires Alves

Resultados da aprendizagem e competências

No fim da unidade curricular o aluno deve ser capaz de:

1. avaliar a eficiência das soluções algorítmicas;
2. desenhar as estruturas de dados de uma aplicação informática de pequena e média complexidade, de acordo com o paradigma de POO.
3. compreender e definir, usando programação genérica, algumas das estruturas de dados abstratas (ADTs) mais importantes;
4. construir as estruturas de dados de baixo nível e respetivos algoritmos que dão suporte dinâmico às ADTs principais.

Pré-requisitos

Antes da unidade curricular o aluno deve ser capaz de: saber programar numa linguagem orientada a objetos.

Conteúdo da unidade curricular

Avaliação de algoritmos; regras elementares de desenho de estruturas de dados; os tipos de dados abstratos (ADTs) lista, pilha, fila, dicionário, árvores binárias de pesquisa e heaps; implementação das estruturas de dados lineares que dão suporte dinâmico à realização dos ADTs.

Conteúdo da unidade curricular (versão detalhada)

1. Análise de algoritmos
 - eficiência algorítmica;
 - análise assintótica;
 - notação big-Oh.
2. Alguns recursos do C# para a implementação de ADTs
 - tipos de dados abstratos (ADTs);
 - classes vs interfaces;
 - classes genéricas;
 - tratamento de exceções.
3. Definição e implementação de estruturas de dados lineares
 - listas simples e duplamente ligadas;
 - stack (pilha) baseada num array;
 - stack baseada numa lista ligada;
 - queue (fila) baseada num array circular;
 - queue baseada numa lista ligada;
 - deque (fila com dupla terminação) baseada num array circular;
 - deque baseada numa lista duplamente ligada;
 - dicionários;
 - Implementação de iteradores.
4. Estruturas em árvore
 - definição de estruturas em árvore;
 - algoritmos de travessia;
 - árvores de decisão;
 - árvores binárias de pesquisa;
 - filas prioritárias baseadas em heaps.

Bibliografia recomendada

1. "Data Structures and Algorithms using C#", Michael McMillan, Cambridge University Press, 2007.
2. "Data Structures and Algorithms: Annotated Reference with Examples", Granville Barnett, and Luca Del Tongo, 2008.
3. "C# 7. 0 com Visual Studio – Curso Completo", Henrique Loureiro, FCA, 2017.

Métodos de ensino e de aprendizagem

As aulas são de cariz teórico-prático, dividindo-se em dois tipos de períodos: - períodos de exposição, durante os quais o docente apresenta os conteúdos, recorrendo de forma intercalada aos métodos expositivos e interrogativos; - períodos de desenvolvimento e implementação. O período não-presencial visa o estudo da matéria dada e a execução de tarefas e trabalhos de cariz prático.

Alternativas de avaliação

1. Alternativa 1 - (Ordinário, Trabalhador) (Final)
 - Trabalhos Práticos - 60%
 - Exame Final Escrito - 40% (componente com nota mínima de 5 valores em vinte)
2. Alternativa 2 - (Ordinário, Trabalhador) (Recurso, Especial)
 - Exame Final Escrito - 100%

Língua em que é ministrada

Português

Validação Eletrónica

Paulo Duarte Ferreira Gouveia	José Luís Padrão Exposto	Paulo Alexandre Vara Alves	Nuno Adriano Baptista Ribeiro
06-03-2020	12-03-2020	20-03-2020	20-03-2020