



Plano de Atividades Domiciliares ADE

Unidade Curricular: Algoritmos e Estruturas de Dados II			
Professor: Álvaro Luiz Fazenda		Contato: alvaro.fazenda@unifesp.br (12) 99133-8633	
Ano Letivo: 2020	Semestre: 1º	Carga horária total: Previsto para o curso presencial: 72 h (46h teórica e 26 prática) Carga já ministrada presencialmente: 8 h Carga horária restante em ADE: 64 h	
Turmas: <i>Integral e Noturna</i>			
Plataforma de acesso ao curso: <i>Moodle UNIFESP</i>			
Objetivos (remoto): Gerais: Introduzir algoritmos e estruturas de dados avançadas através de videoaulas, reuniões virtuais e fóruns de discussão, atividades avaliativas de conteúdo, bem como de práticas de implementação de programas. Ao final do curso o aluno deve estar familiarizado com os algoritmos e estruturas de dados avançadas, devendo ser capaz de definir qual algoritmo e/ou estrutura de dados é mais apropriada para cada tipo de problema. Específicos: Implementar algoritmos e estruturas de dados sofisticadas, para aplicações em memória primária e secundária nas linguagens de programação C, C++ ou Java.			
Conteúdo Programático e Cronograma:			
Sem.	Descrição/Assunto	Carga horária por módulo	Carga horária Sem.
1	Algoritmo de Ordenação <i>Mergesort</i> Vídeo Aula assíncrona Reunião síncrona Atividade Assíncrona via Moodle Atividade de programação assíncrona	1h00m 1h00m 0h30m 2h30m	5h00m
2	Algoritmo de Ordenação <i>Quick Sort</i>		5h00m



	Vídeo Aula assíncrona Reunião síncrona Atividade Assíncrona via Moodle Atividade de programação assíncrona	1h00m 1h00m 0h30m 2h30m	
3	Algoritmo de Ordenação HeapSort Vídeo Aula assíncrona Reunião síncrona Atividade Assíncrona via Moodle Atividade de programação assíncrona	1h00m 1h00m 0h30m 2h30m	5h00m
4	Algoritmos de Ordenação Linear Vídeo Aula assíncrona Reunião síncrona Atividade Assíncrona via Moodle Atividade de programação assíncrona	1h00m 1h00m 0h30m 3h00m	5h30m
5	Busca em Memória Principal - Árvore Binária de Busca AVL (introdução) Vídeo Aula assíncrona Reunião síncrona Atividade Assíncrona via Moodle Atividade de programação assíncrona	2h00m 1h00m 0h30m 2h30m	6h00m
6	Busca em Memória Principal - Árvore AVL - Inserção/Remoção Vídeo Aula assíncrona Reunião síncrona Atividade Assíncrona via Moodle Atividade de programação assíncrona	1h00m 1h00m 0h30m 3h00m	6h30m
7	Busca em Memória Principal - Árvores Rubro-Negra Vídeo Aula assíncrona Reunião síncrona Atividade Assíncrona via Moodle Atividade de programação assíncrona	2h00m 1h00m 0h30m 3h00m	6h30m
8	Busca em Memória Principal - Tabelas Hash - Encadeamento e Endereçamento aberto Vídeo Aula assíncrona Reunião síncrona Atividade Assíncrona via Moodle Atividade de programação assíncrona	2h00m 1h00m 0h30m 3h00m	6h30m
9	Ordenação em memória externa Vídeo Aula assíncrona Reunião síncrona	1h00m 1h00m	5h30m



	Atividade Assíncrona via Moodle Atividade de programação assíncrona	0h30m 3h00m	
10	Pesquisa em Memória Externa Vídeo Aula assíncrona Reunião síncrona Atividade Assíncrona via Moodle Atividade de programação assíncrona	1h00m 1h00m 0h30m 3h00m	5h30m
11	Percurso, menor caminho e busca em largura e profundidade em grafos Vídeo Aula assíncrona Reunião síncrona Atividade Assíncrona via Moodle Atividade de programação assíncrona	2h00m 1h00m 0h30m 3h30m	7h00m
Carga horária total em ADE			64h00

Metodologia de Ensino Utilizada:

Vídeo aulas, reuniões virtuais, fóruns de discussão, atividades avaliativas remotas de conteúdo e práticas de implementação de programas, também de forma remota.

Metodologia de Avaliação (estratégias para atingir conceitos “cumprido” ou “não cumprido”):

Serão realizadas atividades avaliativas remotas assíncronas semanais de duas formas:

- Atividades através do Moodle, Formulários google e demais ferramentas para realização de Quizzes/testes remotos assíncronos.
- Atividades práticas de implementação de programas, de forma remota, a ser submetido pelo sistema Judge disponibilizado pelo campus, e, em algumas circunstâncias submissão pelo Moodle.

Para cada atividade será atribuída uma nota de 0 (zero) a 10 (dez), com escala de uma casa decimal após a vírgula.

A nota final será tomada pela seguinte fórmula:

$$NF = (AM + 2*AP)/3$$

Onde *NF* representa a Nota Final, *AM* (Atividades do Moodle) representa a média aritmética das 12 atividades avaliativas assíncronas (*Quizzes*, testes ou provas) a serem realizadas no Moodle e *AP* (Atividades de Programação) representa a média aritmética das 12 atividades de programação.

Para ter concluído a disciplina o aluno deverá ter $NF \geq 6,0$.

Bibliografia básica e complementar para uso remoto:



A mesma bibliografia já utilizada normalmente, sendo que as referências básicas de número 2 e 3, além das referências complementares 5 e 6, originalmente já citadas e repetidas aqui, encontram-se disponíveis na forma de *e-book*, na biblioteca da Unifesp.

Bibliografia Básica:

- ZIVIANI, N. Projeto de algoritmos: com implementações em PASCAL e C. 2 ed. rev. e ampl. São Paulo: Thomson, 2004. 552 p. ISBN 978-85-221-0390-4 **(Também disponível em e-book)**
- ZIVIANI, Nívio; BOTELHO, Fabiano C. Projeto de algoritmos: com implementações em JAVA e C++. São Paulo: Thomson, 2007. 621 p. ISBN 978-85-221-0525-0 **(Também disponível em e-book).**

Bibliografia Complementar:

- GOODRICH, Michael T et al. Estruturas de dados e algoritmos em Java. Porto Alegre, RS: Bookman, 2007. 600 p. ISBN 978-85-600-3150-4. **(Também disponível em e-book)**
- DROZDEK, Adam. Estrutura de dados e algoritmos em C++. São Paulo: Cengage Learning, c2002. 579 p. ISBN 978-85-221-0295-2. Título original: Data structures and algorithms C++ **(Também disponível em e-book)**

Além das citadas deve-se **acrescentar a seguinte referência acessível online:**

- Khan academy - Computer science - Unit: Algorithms (Dartmouth college professors Tom Cormen and Devin Balkcom). Acessado em: 09/07/2020. <https://www.khanacademy.org/computing/computer-science/algorithms>
-