

CURSO DE ENGENHARIA DE SOFTWARE

1. CARACTERIZAÇÃO DA DISCIPLINA

Nome da Disciplina: Árvores e Grafos	Ano/semestre: 2023/1
Código da Disciplina: 10178	Período: 4º e 5º
Carga Horária Total: 80h/a	Carga Horária Teórica: 60h/a Carga Horária Prática: 20h/a Carga Horária On-line: 00h/a
Pré-Requisito: Estruturas de Dados	Co-Requisito: Não se Aplica

2. PROFESSOR(ES)

Henrique Valle de Lima, M.e

3. EMENTA

Métodos de ordenação: seleção, troca, distribuição, inserção, intercalação e cálculo de endereços. Pesquisa de dados: sequencial, binária, hashing, árvores de pesquisa, árvores binárias de pesquisa, árvores AVL, árvores Patricia, B-Trees. Organização de arquivos.

4. OBJETIVO GERAL

Compreensão de algoritmos clássicos, o domínio das estruturas de dados mais adequadas para a sua implementação, e a percepção crítica sobre quais circunstâncias cada algoritmo é melhor aplicado.

5. OBJETIVOS ESPECÍFICOS

Unidades	Objetivos Específicos
Tipos de ordenação; pesquisa binária e sequencial.	Conhecer e compreender algoritmos clássicos de ordenação e pesquisa.
Tipos de ordenação e tipos de árvores	Compreender a aplicação das estruturas de dados na programação de sistemas computacionais.
Tipos de ordenação, árvores e grafos.	Desenvolver a habilidade de selecionar o algoritmo mais adequado com base nas características do problema a ser tratado.
Tipos de ordenação, árvores e grafos.	Desenvolver posicionamento críticos em relação às vantagens e desvantagens da aplicação das estruturas estudadas.

6. HABILIDADES E COMPETÊNCIAS

Gerais

- G.1. Identificar problemas que tenham solução algorítmica;
- G.2. Conhecer e compreender os limites da computação;
- G.3. Tomar decisões, avaliando criticamente, soluções computacionais, consciente dos aspectos humanos, éticos, legais e ambientais decorrentes;
- G.4. Gerir a própria aprendizagem e desenvolvimento pessoal-profissional, realizando trabalho em equipe, com visão trans e interdisciplinar;
- G.5. Desenvolver trabalhos e soluções, adotando metodologias diversificadas;
- G.8. Resolver problemas usando ambientes de programação.

Específicas

E.1. Investigar, compreender e estruturar as características de domínios de aplicação em diversos contextos para a construção de sistemas de software, considerando questões éticas, sociais, legais e econômicas, individualmente e/ou em equipe;

E.2. Compreender e aplicar processos, técnicas e procedimentos de construção inerentes à produção e utilização de software, conhecendo os direitos e propriedades intelectuais;

1. E.7. Analisar e criar novos modelos no desenvolvimento de software, identificando oportunidades e desenvolvendo soluções inovadoras.

7. CONTEÚDO PROGRAMÁTICO

Semana	Data	Conteúdo	Estratégia de ensino-aprendizagem	Aula Teórica/Prática	Local
1	16/02	Apresentação do Plano de ensino; Revisão de programação.	- Aula expositiva - Atividade pós-aula	Teórica	- Sala de Aula - Ambiente Virtual de Aprendizagem
2	23/02	Ordenação por troca: Bubblesort; Ordenação por inserção: ShellSort; Insertion Sort. Exercícios em sala e para casa.	- Aula expositiva - Atividade pós-aula	Teórica	- Sala de Aula - Ambiente Virtual de Aprendizagem
3	02/03	Ordenação por troca: Quick Sort; Ordenação por seleção: Seleção direta e Heapsort;	- Leitura de referência bibliográfica - Aula expositiva - Atividade pós-aula	Teórica	- Sala de Aula - Ambiente Virtual de Aprendizagem
4	09/03	Ordenação por troca: Quick Sort; Ordenação por seleção: Seleção direta e Heapsort;	- Aula expositiva - Atividade pós-aula - Atividade prática laboratorial	Prática	- Laboratório de Informática - Ambiente Virtual de Aprendizagem
5	16/03	Ordenação por seleção: Heapsort;	- Leitura de referência bibliográfica - Aula expositiva - Atividade pós-aula	Teórica	- Sala de Aula - Ambiente Virtual de Aprendizagem
6	23/03	Ordenação por intercalação: Mergesort	- Aula expositiva - Atividade pós-aula - Atividade prática laboratorial	Teórica	- Laboratório de Informática - Ambiente Virtual de Aprendizagem
7	30/03	1ª Verificação de Aprendizagem	- Prova individual	Teórica	- Sala de Aula
8	06/04	Devolutiva qualificada da 1ª V.A.	- Devolutiva qualificada	Teórica	- Sala de Aula
9	13/04	Algoritmos de pesquisa: pesquisa sequencial e binária. Árvore AVL	- Leitura de referência bibliográfica - Aula expositiva - Atividade pós-aula	Teórica	- Sala de Aula - Ambiente Virtual de Aprendizagem
10	20/04	Algoritmos de pesquisa: pesquisa sequencial e binária. Árvore AVL	- Leitura de referência bibliográfica - Aula expositiva - Atividade pós-aula	Teórica	- Sala de Aula - Ambiente Virtual de Aprendizagem
11	27/04	Árvore AVL.	- Atividade pós-aula - Atividade prática laboratorial	Prática	- Laboratório de Informática - Ambiente Virtual de Aprendizagem
12	04/05	Árvore Patricia. B-tree.	- Leitura de referência bibliográfica - Aula expositiva - Atividade pós-aula - Atividade prática laboratorial	Teórica e Prática	- Laboratório de Informática - Ambiente Virtual de Aprendizagem
13	11/05	B-tree e Hashing.	- Leitura de referência bibliográfica - Aula expositiva - Atividade pós-aula - Atividade prática laboratorial	Teórica e Prática	- Laboratório de Informática - Ambiente Virtual de Aprendizagem
14	18/05	2ª Verificação de Aprendizagem	- Prova individual	Teórica	- Sala de Aula
15	25/05	Devolutiva qualificada da 2ª V.A.	- Devolutiva qualificada	Teórica	- Sala de Aula
16	01/06	Conceito e Introdução de grafos.	- Leitura de referência bibliográfica - Aula expositiva - Atividade pós-aula - Atividade prática laboratorial	Teórica e Prática	- Laboratório de Informática - Ambiente Virtual de Aprendizagem
17	15/06	Grafos ARP	- Atividade pós-aula - Atividade prática laboratorial	Prática	- Sala de Aula - Ambiente Virtual de Aprendizagem
18	22/06	Grafos	- Atividade pós-aula	Prática	- Sala de Aula

			- Atividade prática laboratorial		- Ambiente Virtual de Aprendizagem
19	22/06	Correção ARP e Revisão de conteúdo	- Atividade pós-aula - Atividade prática laboratorial	Prática	- Sala de Aula
20	05/07	3ª Verificação de aprendizagem	- Prova individual	Teórica	- Sala de Aula

* As VERIFICAÇÕES DE APRENDIZAGEM podem ser aplicadas de forma presencial ou virtual, bem como ter suas datas alteradas a depender do quadro epidemiológico da pandemia da COVID19.

8. PROCEDIMENTOS DIDÁTICOS

Atividade avaliativa, aula expositiva dialogada, retomada de conteúdo, problematização, mapa conceitual, seminário, trabalho em grupo e Tecnologias da Informação e Comunicação – vídeos, mapa mental, AVA – plataforma Moodle com Vídeo do YouTube

Recursos de Acessibilidade disponíveis aos acadêmicos

O curso assegura acessibilidade metodológica, digital, comunicacional, atitudinal, instrumental e arquitetônica, garantindo autonomia plena do discente.

9. ATIVIDADE INTEGRATIVA

A interdisciplinaridade no curso de Engenharia de Software é construída com o amparo das disciplinas de Projeto Interdisciplinar. Estas promovem a associação entre os diferentes conteúdos, habilidades e cenários em projetos que favoreçam a construção do conhecimento científico, tecnológico e de prática profissional aliado à autoaprendizagem, proatividade, resolução conjunta de problemas, trabalho em equipe, reflexividade, entre outros.

A proposta de cada disciplina de Projeto Interdisciplinar é variável, mas, obrigatoriamente, deve evoluir em uma constante de maturidade pessoal, interpessoal, científica e prática. Para o desenvolvimento dos projetos interdisciplinares o aluno percorre três momentos: ensino – por meio do diálogo entre as áreas de conhecimento; pesquisa – seguindo os rigores metodológicos necessários à construção do conhecimento científico e de extensão – oportunizando o compartilhamento dos projetos desenvolvidos para o público interno e externo.

Em função disto, as atividades de cada Projeto estão detalhadas em Plano de Ensino próprio.

10. PROCESSO AVALIATIVO DA APRENDIZAGEM

1ª Verificação de aprendizagem (V. A.) – valor 0 a 100 pontos

Avaliação com valor 0 a 50 pontos.

Avaliações processuais totalizam 0 a 50 pontos distribuídos da seguinte forma:

- **APS: 9 Pts divididos em 6 atividades – 1,5 por APS**
- **APA: 6 Pts divididos em 6 atividades – 1 pt por APA**
- **Aprendendo a Resolver Problemas - ARP: 10 pts em uma atividade.**
- **Atividade Prática: 15 pts divididos em 5 atividades – 3 pts por Atividade**
- **Projeto Integrativo: 10 Pontos.**

A média da 1ª V. A. será a somatória da nota obtida na avaliação teórica (0 a 50 pontos) e as notas obtidas nas avaliações processuais (0 a 50 pontos).

(a devolutiva será realizada conforme Cronograma).

2ª Verificação de aprendizagem (V. A.) – valor 0 a 100 pontos

Ex.: Avaliação com valor 0 a 50 pontos.

Avaliações processuais totalizam 0 a 50 pontos distribuídos da seguinte forma:

- **APS: 9 Pts divididos em 6 atividades – 1,5 por APS**
- **APA: 6 Pts divididos em 6 atividades – 1 pt por APA**
- **Aprendendo a Resolver Problemas - ARP: 10 pts em uma atividade.**
- **Atividade Prática: 15 pts divididos em 5 atividades – 3 pts por Atividade**
- **Projeto Integrativo: 10 Pontos.**

A média da 2ª V. A. será a somatória da nota obtida na avaliação teórica (0 a 50 pontos) e a nota obtida nas avaliações processuais (0 a 50 pontos).

(a devolutiva será realizada conforme Cronograma).

3ª Verificação de aprendizagem (V. A.) – valor 0 a 100 pontos

Ex.: Avaliação com valor 0 a 50 pontos.

Avaliações processuais totalizam 0 a 50 pontos distribuídos da seguinte forma:

- **APS: 7,5 Pts divididos em 5 atividades – 1,5 por APS**
- **APA: 5 Pts divididos em 5 atividades – 1 pt por APA**
- **Aprendendo a Resolver Problemas - ARP: 10 pts em uma atividade.**
- **Atividade Prática: 12 pts divididos em 4 atividades – 3 pts por Atividade**
- **Projeto Integrativo: 15,5 Pontos.**

A média da 3ª V. A. será a somatória da nota obtida na avaliação teórica (50 pontos) e as notas obtidas nas avaliações processuais (50 pontos). (a devolutiva será realizada conforme Cronograma).

ORIENTAÇÕES ACADÊMICAS

- Nas três VAs – O pedido para avaliação substitutiva tem o prazo de 3 (três) dias úteis a contar da data de cada avaliação com apresentação de documentação comprobatória (§ 1º e § 2º do art. 39 do Regimento Geral do Centro Universitário UniEVANGÉLICA). **A solicitação deverá ser feita através do Sistema Acadêmico Lyceum obrigatoriamente.**
- Nas três VAs – O pedido para Revisão de nota tem o prazo de 3 (três) dias úteis a contar da data da publicação, no Sistema Acadêmico Lyceum, do resultado de cada avaliação. (Art. 40 do Regimento Geral do Centro Universitário UniEVANGÉLICA).
- Atribui-se nota zero ao aluno que deixar de submeter-se às verificações de aprendizagem nas datas designadas, bem como ao que nela se utilizar de meio fraudulento. (Capítulo V Art. 39 do Regimento Geral do Centro Universitário UniEVANGÉLICA)

Participação em eventos científicos:

Portaria – Frequência e nota dos alunos que apresentarem trabalhos em eventos científicos

Seguir as orientações presentes na Portaria Nº 01, de 7 de fevereiro de 2019, dos Bacharelados em Computação, que dispõe sobre os procedimentos de justificativa de ausência para alunos que apresentarem trabalhos em eventos científicos.

Condição de aprovação

Considera-se para aprovação do (a) acadêmico (a) na disciplina, frequência mínima igual ou superior a 75% da carga horária e nota igual ou superior a sessenta (60) obtida com a média aritmética simples das três verificações de aprendizagem.

11. BIBLIOGRAFIA

Básica:

CELES, Valdemar et al. Introdução a Estrutura de Dados com Técnicas de Programação em C. 2ª ed. Ed. Elsevier, 2016.
EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS, Brasil: Bookman, 2009. 262 p. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788577804504>
TENENBAUM, Aaron M.; LANGSAM, Yeddyah; AUGENSTEIN, Moshe J.. Estruturas de Dados usando C. São Paulo: Pearson Education, 2013.

Complementar:

KOFFMAN, Elliot B., WOLFGANG, Paul T. Objetos, Abstração, Estrutura de Dados e Projeto Usando C++. Rio de Janeiro: LTC, 2008. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/978-85-216-2780-7>
MIZRAHI, Victorine Viviane. Treinamento em linguagem C. São Paulo, Makron Books, 2008
ASCENCIO, A. F. G. et al. Estrutura de Dados: algoritmos, análise de complexidade e implementações em JAVA e C/C++. São Paulo: Pearson Prentice Hall, 2010.
SEBESTA, Robert W.. Conceitos de linguagens de programação. 5. ed. Porto Alegre, RS, Brasil: Bookman, 2003. 638 p.
ZIVIANI, Nivio. Projeto de algoritmos: com implementações em Pascal e C. 3. ed. São Paulo: Cengage Learning, 2011. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788522126590>.

Anápolis, 09 de fevereiro de 2023.




Prof. M.e Natasha Sophie Pereira

COORDENADOR (A) DO CURSO DE ENGENHARIA DE SOFTWARE DA UniEVANGÉLICA



Prof. M.e William Pereira dos Santos Júnior

COORDENADOR(A) PEDAGÓGICO(A) DO CURSO DE ENGENHARIA DE SOFTWARE DA UniEVANGÉLICA



Prof. M.e William Pereira dos Santos Júnior
PROFESSOR(A) RESPONSÁVEL PELA DISCIPLINA

