

CURSO:	Engenharias	SEMESTRE/ANO:	2025/2
DISCIPLINA:	Estruturas de Dados 2	CRÉDITOS:	04
CARGA HORÁRIA:	60 horas		
PROFESSORES:	John Lenon Cardoso Gardenghi Edson Alves da Costa Júnior		

PLANO DE ENSINO

1 Objetivos da Disciplina

Tornar o aluno capaz de implementar problemas de maior sofisticação técnica, utilizando estruturas de dados não-lineares, como árvores e grafos, e algoritmos de ordenação eficientes, os quais são recorrentes em situações de ordem prática e em sistemas reais.

2 Ementa do Programa

I. *Estruturas não-lineares*

- i. Árvores
- ii. Tabelas *hash*
- iii. Filas de prioridade
- iv. *Heaps*

II. *Algoritmos de Ordenação*

- i. Algoritmos de Ordenação $O(n \log n)$

- ii. Algoritmos de Ordenação $O(n)$

III. *Grafos*

- i. Representação de grafos
- ii. Algoritmos em grafos
- iii. Aplicações

3 Horário das aulas e atendimento

AULAS: terças e quintas, das 14:00 às 15:50 hrs.

ATENDIMENTO: segundas, das 12:30 às 14:30 hrs, via plataforma Teams.

4 Metodologia

A metodologia consiste em aulas expositivas, com o auxílio do quadro branco e projetor digital. A fim de fortalecer a aprendizagem da disciplina, as aulas serão complementadas com exercícios e atividades, presenciais e extra-classe.

Também serão utilizadas listas de exercícios para prática dos conceitos apresentados em aula e para preparação para as avaliações.

5 Critérios de Avaliação

5.1 Provas

A avaliação do curso será feita através de duas provas teóricas e duas provas práticas.

5.1.1 Provas Teóricas

As provas teóricas T_1 e T_2 serão compostas por questões de múltipla escolha e valerão 10 pontos cada. O estudante deve marcar, **à caneta**, uma, e apenas uma, dentre as alternativas apresentadas para cada questão. Uma questão será considerada certa apenas se o estudante marcou a alternativa correta, de acordo com o gabarito. Serão consideradas incorretas questões sem marcação, com mais de uma marcação, com rasuras ou com marcações à lapis.

5.1.2 Provas Práticas

As provas práticas P_1 e P_2 serão compostas por problemas a serem resolvidos na plataforma MOJ. Serão permitidas submissões de soluções escritas em C, C++ ou Python.

Uma solução proposta para cada problema será corrigida de acordo com os seguintes critérios: após ser **compilado** ou **interpretado** de forma bem sucedida, uma série de **testes unitários automatizados** alimentarão o programa com **entradas válidas** e comparará os resultados obtidos com as **saídas corretas**. Uma solução problema será considerada **correta** se obtiver sucesso em **todos os testes unitários**.

5.2 Listas de exercícios

Serão propostas listas, na plataforma MOJ, com exercícios relacionados com o conteúdo ministrado. A resolução das listas não modifica a menção, mas é fortemente encorajada para a fixação dos conceitos apresentados no curso.

5.3 Atividades Extras

Poderão ser aplicadas, a critério do professor, atividades extras, de caráter facultativo. A forma, data de entrega e método de avaliação de tais atividades serão divulgados na plataforma SIGAA. A pontuação atribuída a tais atividades será somada à nota final do aluno.

5.4 Nota

A nota N do aluno será dada pela expressão por 10:

$$N = \frac{2(T_1 + T_2) + 3(P_1 + P_2)}{10}$$

5.5 Menção Final

A menção final do curso será dada em função da nota N , de acordo com a tabela abaixo.

N	Menção	Descrição
0	SR	<i>Sem rendimento</i>
0,1 a 2,9	II	<i>Inferior</i>
3,0 a 4,9	MI	<i>Médio inferior</i>
5,0 a 6,9	MM	<i>Médio</i>
7,0 a 8,9	MS	<i>Médio superior</i>
9,0 a 10	SS	<i>Superior</i>

No final do semestre será aplicada uma **prova substitutiva**, individual, caso o aluno apresente um atestado de saúde em até 5 (cinco) dias após a realização da prova, ou em outros casos previstos em lei (alistamento militar, etc). A prova substitutiva corresponderá à avaliação perdida pelo aluno e abrangerá todo o conteúdo do curso.

5.6 Critérios de aprovação

Obterá **aprovação** no curso o aluno que cumprir as **duas** exigências abaixo:

1. Ter presença em 75% ou mais das aulas;
2. Obter menção igual ou superior a MM.

IMPORTANTE: Atestados médicos e documentos comprobatórios de justificativa de faltas dão direito à realização de atividades avaliativas que você venha a perder, mas essas ausências justificadas também são levadas em consideração como ausências efetivas para o cômputo da frequência mínima obrigatória (*Graduação UnB – Manual para estudantes*, pág. 35).

6 Cronograma

Semana	Aula	Data	Conteúdo
01	1	19/08	<i>Apresentação do curso. Recursão.</i>
	2	21/08	<i>Exercícios de recursão.</i>
02	3	26/08	<i>Árvores binárias. Árvores de busca binárias.</i>
	4	28/08	<i>Inserção, remoção, busca e travessia em árvores de busca binárias.</i>
03	5	02/09	<i>Balanceamento de árvores.</i>
	6	04/09	<i>Árvores Red-Black: Parte I.</i>
04	7	09/09	<i>Árvores Red-Black: Parte II.</i>
	8	11/09	<i>Heap Binária: Parte I.</i>
05	9	16/09	<i>Heap Binária: Parte II.</i>
	10	18/09	<i>Árvores m-árias. Implementação de árvores.</i>
06	11	23/09	<i>Definição de hashes.</i>
	12	25/09	<i>Sondagem linear e quadrática.</i>
07	-	30/09	Prova Teórica 1
	13	02/10	<i>Árvore de Fenwick.</i>
08	14	07/10	<i>Árvore de Segmentos: Parte I.</i>
	15	09/10	<i>Árvore de Segmentos: Parte II.</i>
09	16	14/10	<i>Definição de grafos.</i>
	17	16/10	<i>DFS.</i>
10	18	21/10	<i>BFS.</i>
	19	23/10	<i>Componentes Conectados.</i>
11	20	28/10	<i>Union-Find Disjoint Sets.</i>
	21	30/10	<i>Ordenação Topológica.</i>
12	-	04/11	Semana de Extensão Universitária
	-	06/11	Semana de Extensão Universitária
13	22	11/11	<i>Grafos bipartidos.</i>
	23	13/11	<i>Pontes e pontos de articulação.</i>
14	-	18/11	Feriado: Dia de Zumbi e Consciência Negra
	24	20/11	<i>Árvore geradora mínima.</i>
15	25	25/11	<i>Algoritmo de Dijkstra.</i>
	-	27/11	Prova Teórica 2
16	-	02/12	Prova Prática 1
	-	04/12	Prova Prática 2
17	-	09/12	Prova substitutiva
	-	11/12	Menções finais

7 Bibliografia

Todos os livros abaixo se encontram disponíveis na Biblioteca da FGA, na Biblioteca Virtual da UnB ou tem acesso livre.

LIVRO TEXTO

CORMEN, Thomas H., **LEISERSON**, Charles E., **RIVEST**, Ronald L, **STEIN**, Clifford. *Algoritmos: Teoria e Prática*, Elsevier, 2002.

DROZDEK, Adam. *Estruturas de Dados e Algoritmos em C++*, Thomson, São Paulo, 2002.

LAFORE, R. *Estruturas de Dados e Algoritmos em Java*, 1ª edição, Ciência Moderna, 2005.

LITERATURA COMPLEMENTAR

CORMEN, Thomas H., **LEISERSON**, Charles E., **RIVEST**, Ronald L, **STEIN**, Clifford. *Introduction to Algorithms*, MIT Press, 2014. (eBrary)

MEHLHORN, K., **SANDERS**, P. *Algorithms and Data Structures: The Basic ToolBox*, 1st edition, Springer, 2008. (eBrary)

HALIM, Steve S., **HALIM**, Felix. *Competitive Programming*, 1st edition, Lulu, 2010. (Open Access)

STEPHENS, Rod. *Essential Algorithms: A Practical Approach to Computer Algorithms*, John Wiley & Sons, 2013. (eBrary)

AHO, A. V., **ULLMAN**, J. D.. *Foundations of Computer Science: C Edition*, 1st edition, W. H. Freeman, 1994. (Open Access).