



UNIVERSIDADE FEDERAL DE OURO PRETO  
DEPARTAMENTO DE COMPUTAÇÃO

PLANO DE ENSINO



<b>Nome do Componente Curricular em português:</b> Estruturas de Dados I		<b>Código:</b> BCC202
<b>Nome do Componente Curricular em inglês:</b> Data Structures I		
<b>Nome e sigla do departamento:</b> Departamento de Computação (DECOM)		<b>Unidade acadêmica:</b> ICEB
<b>Nome do docente:</b> Pedro Henrique Lopes Silva e Karla Alexsandra de Souza Joriatti		
<b>Carga horária semestral:</b> 90 horas	<b>Carga horária semanal teórica:</b> 4 horas/aula	<b>Carga horária semanal prática:</b> 2 horas/aula
<b>Data de aprovação na assembleia departamental:</b> 29/08/2025		
<b>Ementa:</b> Recursividade; conceitos básicos de análise assintótica de algoritmos; tipos abstratos de dados; estruturas de dados: listas, pilhas, filas de prioridade e árvores binárias; algoritmos de ordenação por comparação de chaves: seleção, inserção, bolha, shellsort, quicksort, mergesort, heapsort; algoritmos de ordenação em tempo linear: counting sort, radix sort e bucket sort; e algoritmos de pesquisa: simples, binária, árvores binárias de busca e hashing.		
<b>Conteúdo Programático:</b> <ul style="list-style-type: none"><li>• Revisão de alocação dinâmica de memória</li><li>• Recursividade</li><li>• Noções de análise de complexidade de algoritmos:</li><li>• Conceitos</li><li>• Medidas de avaliação: tempo e espaço</li><li>• Análise assintótica: notação O, Omega e Theta</li><li>• Hierarquia de funções e classes de problemas</li><li>• Tipos de dados abstratos</li><li>• Estruturas de Dados</li><li>• Listas</li><li>• Pilhas</li><li>• Filas</li><li>• Filas de prioridade</li><li>• Árvores</li><li>• Algoritmos</li><li>• Métodos de ordenação por comparação: Selection Sort, Insertion Sort, Bubblesort, Shellsort, Quicksort, Heapsort e Mergesort</li><li>• Métodos de ordenação em tempo linear: Counting Sort, Radix Sort e Bucket Sort</li><li>• Métodos de pesquisa: Simples, Binária, Árvores Binárias e AVL e Hashing</li></ul>		

**Objetivos:**

O aluno deverá conhecer conceitos associados a tipos abstratos de dados e métodos de pesquisa e ordenação de interesse teórico e prático.

Deverá também adquirir a capacidade de utilizar esses recursos pra desenvolvimento de programas, utilizando conceitos de modularização e abstração de dados.

Deverá ainda ser capaz de comparar estratégias de implementação do ponto de vista da complexidade dos algoritmos envolvidos, usando a notação O.

**Metodologia:**

**Aulas presenciais:** As atividades envolvem:

- Preparação de aulas e outros tipos de materiais complementares.
- Leituras recomendadas de textos técnicos, apostilas e livros de autores especialistas com a finalidade de proporcionar ao discente a oportunidade de consulta e desenvolvimento de sua capacidade de análise, síntese e crítica de uma bibliografia específica.
- Aulas práticas (P) com a implementação dos métodos estudados. O professor proverá auxílio durante o horário da aula prática. Exercícios resolvidos deverão ser enviados em formato de código-fonte pelo o ambiente virtual de aprendizagem adotado.
- Trabalhos práticos abrangendo todo o conteúdo da disciplina. Os alunos deverão entregar o trabalho pelo ambiente virtual de aprendizagem adotado e apresentá-lo para o(s) professor(es) em um horário específico se assim for solicitado.
- Não é permitido o uso de celulares e computadores durante as aulas teóricas.

**Recursos utilizados:** Para o desenvolvimento da aprendizagem serão adotados, concomitantemente e em todos os tópicos da disciplina, os seguintes recursos de apoio didático-pedagógico:

- Ferramentas do ambiente virtual de aprendizagem adotado (Moodle e corretor automático).
- Computador desktop ou notebook.
- Livros, Apostila, textos científicos e documentos legais em formato impresso ou digital (disponíveis gratuitamente online) para estudos extraclasse.
- As implementações das aulas práticas e trabalhos práticos necessitam de um ambiente de programação e compilação. O aluno pode optar por instalar as ferramentas necessárias (compilador e editor de texto), que são gratuitas, ou utilizar um ambiente web que não requer nenhuma instalação.

**Atividades avaliativas:**

- Três Provas Teóricas de 10 pontos.
- Três Trabalhos Práticos (TPs) de 10 pontos.
- Avaliações Contínuas (AC) de 10 pontos.

Média Final =  $(2,0 * Prova I + 2,0 * Prova II + 2,5 * Prova III + 1 * TP I + 1 * TP II + 1 * TP III + 0,5 * AC) / 10$

- Provas Teóricas: os alunos receberão uma prova com questões referentes aos conteúdos estudados e terão o horário da aula para fazer e entregar a prova. As provas serão aplicadas nos horários cadastrados das disciplinas/turmas.
- Avaliações Contínuas: aulas práticas referentes aos conteúdos estudados distribuídos ao longo do semestre e atividades teóricas ao longo do período.
- Frequência: a computação da frequência será mediante a presença em aula.

- Exame Especial: O Exame Especial será uma prova única e individual. Cada aluno receberá uma prova com questões referentes aos conteúdos estudados e terão o tempo da aula para resolver e entregar.

**Resolução CEPE 2880 de 05/2006:** É assegurado a todo aluno regularmente matriculado com frequência mínima de **setenta e cinco por cento** e média inferior a seis, o direito de ser avaliado por Exame Especial.

#### **Cronograma:**

Atividades propostas:

Semana 1	-	07/10/25	-	Apresentação do Curso
	-	09/10/25	-	Princípios da Programação em C
Semana 2	-	14/10/25	-	Alocação Dinâmica
	-	16/10/25	-	Tipos Abstratos de Dados (TADs)
Semana 3	-	21/10/25	-	Recursividade
	-	23/10/25	-	Análise Assintótica de Algoritmos I
Semana 4	-	28/10/25	-	Recesso Acadêmico
	-	30/10/25	-	Análise Assintótica de Algoritmos II
Semana 5	-	04/11/25	-	Análise Assintótica de Algoritmos III
	-	06/11/25	-	Recursividade e Análise Assintótica
Semana 6	-	11/11/25	-	Listas
	-	13/11/25	-	Listas Encadeadas
Semana 7	-	18/11/25	-	Prova I e Entrega do TP I
	-	20/11/25	-	Recesso Acadêmico
Semana 8	-	25/11/25	-	Pilhas
	-	27/11/25	-	Filas
Semana 9	-	02/12/25	-	Métodos de Ordenação
	-	04/12/25	-	MergeSort
Semana 10	-	09/12/25	-	QuickSort
	-	11/12/25	-	ShellSort
Semana 11	-	16/12/25	-	Prova II e Entrega do TP II
	-	18/12/25	-	Fila de Prioridade
Semana 12	-	20/01/26	-	HeapSort
	-	22/01/26	-	Ordenação em Tempo Linear
Semana 13	-	27/01/26	-	Pesquisa Sequencial e Pesquisa Binária
	-	29/01/26	-	Árvores Binárias de Pesquisa
Semana 14	-	03/02/26	-	Árvores Binárias de Pesquisa II
	-	05/02/26	-	Árvores AVL
Semana 15	-	10/02/26	-	Tabelas Hash por Lista Encadeada
	-	12/02/26	-	Tabelas Hash Endereçamento Aberto
Semana 16	-	17/02/26	-	Recesso Acadêmico
	-	19/02/26	-	Prova III e Entrega do TP III
Semana 17	-	24/02/26	-	Correção da Prova
	-	26/02/26	-	Preparação para o Exame Especial
Semana 18	-	03/03/26	-	Exame Especial

**Bibliografia Básica:**

- ZIVIANI, Nivio. Projeto de algoritmos: com implementações em Pascal e C. 3. ed. rev. e ampl. São Paulo: Cengage Learning, c2011. xx, 639 p. ISBN 9788522110506.
- CELES, Waldemar; CERQUEIRA, Renato; RANGEL, Jose Lucas. Introdução a Estruturas de Dados: com técnicas de programação em C.. Rio de Janeiro: Elsevier 2004. 293 p.
- CORMEN, Thomas H. Algoritmos: teoria e prática. Rio de Janeiro: Campus, 2002. xvii, 916 p.

**Bibliografia Complementar:**

- KLEINBERG, Jon; TARDOS, Eva. Algorithm design. Boston: Pearson/Addison-Wesley, c2006. xxiii, 838 p.
- KNUTH, Donald Ervin. The art of computer programming. Upper Saddle River: Addison Wesley, c2005. v.
- GOODRICH, Michael T.; TAMASSIA, Roberto; COPSTEIN, Bernardo. Projeto de algoritmos: fundamentos, análise e exemplos da internet. Porto Alegre: Bokman, 2004. 696 p.
- DROZDEK, Adam. Estrutura de dados e algoritmos em C++. São Paulo: Cengage Learning, c2002. xviii, 574 p.
- TENENBAUM, Aaron M; LANGSAM, Yedidyah; AUGENSTEIN, Moshe. Estruturas de dados usando C. São Paulo: Makron Books, 1995. 884 p.