

Plano de Ensino e Aprendizagem

1) Identificação

| | |
|-----------------|-------------------------------------|
| Curso | (417) Ciência da Computação - Manhã |
| Carga horária | 80 h |
| Disciplina | (G01P2) Estruturas de Dados |
| Semestre letivo | 2026/1 - Graduação |
| Professor | Alexandre de Oliveira Zamberlan |

2) Ementa

Estruturas lineares; Matrizes; Árvores; Grafos.

3) Objetivo da Disciplina – Competências

Unidade 1: Estruturas lineares

Criar estruturas de dados lineares utilizadas em programação de maneira útil.

Unidade 2: Matrizes

Manipular as diferentes matrizes especiais na construção de programas de computador para melhor otimização de memória.

Unidade 3: Árvores

Criar estruturas de dados não lineares do tipo árvores utilizadas em programação de maneira útil.

Unidade 4: Grafos

Gerar estruturas de dados não lineares do tipo grafo utilizadas em programação de maneira útil.

4) Abertura da Disciplina

Na disciplina de Estrutura de Dados estudaremos as principais estruturas de dados abstratas e seus algoritmos de manipulação (inserção, remoção, atualização, busca e percurso), bem como suas principais aplicações em situações problemas do dia-a-dia, em soluções usando as linguagens C, C++, C#, Python e Java. Nesta disciplina, aprenderemos e aplicaremos boas práticas de desenvolvimento de sistemas em situações de problemas do mundo real.

5) Caracterização da metodologia de ensino

A disciplina será presencial em laboratório de informática e com apoio da ferramenta TEAMS (em aulas e situações específicas). Os conteúdos serão apresentados de forma expositivo-dialogados e com práticas nas linguagens C, C++, Java, Python e C#, instaladas nos computadores dos alunos e/ou nas máquinas do laboratório. O repositório GITHUB será usado para armazenar os programas desenvolvidos. Já a ferramenta TEAMS funcionará como mecanismo de comunicação entre professor e aluno em aulas específicas a combinar.

6) Avaliação da aprendizagem

Serão critérios de avaliação:

- capacidade de contextualizar teoria à prática;
- construção de soluções diversificadas aos problemas propostos;
- envolvimento nos trabalhos;

- iv) envolvimento nas aulas com participação crítica ao conteúdo e aos exercícios apresentados;
- v) funcionamento correto dos programas, além da qualidade da implementação e sua eficiência.
- vi) provas e trabalhos: análise e discussão sobre soluções apresentadas.

A nota será aferida por meio de três avaliações parciais. A primeira avaliação será referente ao conteúdo de estruturas lineares; a segunda envolverá os conteúdos de matrizes esparsas e árvores binárias, e a terceira os conteúdos de árvores binárias de pesquisa e grafos.

As notas 1, 2 e 3 serão obtidas da seguinte forma:

$N1 = (\text{Produto de aprendizagem 1 } 0.6) + (\text{Exercícios } 0.3) + (\text{Participação efetiva } * 0.1)$

$N2 = (\text{Produto de aprendizagem 2 } 0.6) + (\text{Exercícios } 0.3) + (\text{Participação efetiva } * 0.1)$

$N3 = (\text{Produto de aprendizagem 3 } 0.6) + (\text{Exercícios } 0.3) + (\text{Participação efetiva } * 0.1)$

A nota final da disciplina será obtida por meio da média aritmética simples entre as notas N1, N2 e N3 conforme cálculo:

$\text{Nota Final} = (N1 + N2 + N3) / 3$

É possível que, durante o semestre, seja verificada a possibilidade de realizar exercícios ou trabalhos de recuperação. Neste caso, a nota do exercício/trabalho ajudará a compor uma nota parcial com a próxima prova escrita, o que será avisado aos alunos.

Para o aluno ser aprovado precisará de, no mínimo, 75% de frequência às aulas. Aluno com média semestral igual ou superior a 6.0 estará aprovado, caso contrário, reprovado. A participação em aula poderá ser utilizada como critério de arredondamento das notas.

Serão aceitas somente as justificativas de ausências em avaliações teóricas previstas pela Central do Aluno e apresentadas até 48 horas após a ocorrência.

7) Bibliografia básica

EDELWEISS, Nina; GALANTE, Renata. **Estruturas de dados**. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18). ISBN 978-85-7780-381-1.

Forbellone, André Luiz Villar; Eberspacher, Henri Frederico. **Lógica de programação: A construção de algoritmos e estruturas de dados com aplicações em Python**, 2022. (Biblioteca Digital)

Vinicius Pozzobon Borin. **Estrutura de dados**, 2020. (Biblioteca Digital)

8) Bibliografia complementar

GOODRICH, Michael T.; TAMASSIA, Roberto. **Estruturas de dados e algoritmos em Java**. 4. ed. Porto Alegre: Bookman, 2007. xiii, 600 p. ISBN 978-85-60031-50-4

KNUTH, Donald E. **The art of computer programming**. 3rd. ed. Boston: Addison Wesley, c1997. V. 1

SZWARCFITER, Jayme Luiz; MARKENZON, Lilian. **Estruturas de dados e seus algoritmos**. 2. ed. rev. 1994. Rio de Janeiro, RJ: LTC, 1994. 320 p.

ZIVIANI, Nivio. **Projeto de algoritmos com implementações em Pascal e C**. 5. ed. São Paulo, SP: Pioneira, 2002. 267 p.

CORMEN, Thomas H. **Algoritmos: teoria e prática**. Rio de Janeiro, RJ: Campus, 2002. 916 p.

Roteiro de estudos

1º aula / 1º semana

Modalidade da aula: Presencial - 18 de February de 2026

Unidade 1 - Estruturas lineares

- Experiência formativa

O que estudaremos?

Estudaremos a implementação de listas e estruturas lineares com restrições: pilhas, filas e DEQUES de maneira útil.

O que aprenderemos?

Aprenderemos como compor estruturas de dados na construção de programas de computador, como desenvolver algoritmos de estruturas de dados lineares para problemas específicos na área da computação, além disso como combinar soluções computacionais que utilizem estruturas de dados lineares.

Sobre o que refletiremos?

Como aplicar as estruturas de dados lineares em problemas reais? Como implementar as estruturas de dados nas linguagens C, C++, Java e C#? Como as estruturas de dados lineares são fundamentais em outras áreas da Computação?

- Competências

Criar estruturas de dados lineares utilizadas em programação de maneira útil.

- Objetivos

- 1.1 Compor estruturas de dados na construção de programas de computador;
- 1.2 Desenvolver algoritmos de estruturas de dados lineares para problemas específicos na área da computação;
- 1.3 Combinar soluções computacionais que utilizem estruturas de dados lineares.

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os algoritmos de inserção, remoção, atualização e percurso de Pilhas Encadeadas.

- Guia de Aprendizagem

Nesta aula vamos discutir o Plano de Ensino, as diferenças entre estruturas de dados físicas e abstratas, os tipos e as aplicações de estruturas de dados abstratas. Além disso, vamos implementar os métodos de operação em Pilha Encadeada.

- Encaminhamento próxima aula - Atividade Discente Efetiva

Pesquise sobre o ambiente GitHub (versionamento de código), crie conta, clone o repositório da disciplina, crie seu repositório pessoal da disciplina.

- Referência Bibliográfica

EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18).

2º aula / 2º semana

Modalidade da aula: Presencial - 25 de February de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os algoritmos de inserção, remoção, atualização e percurso de Pilhas Encadeadas. Utilizar na programação arquivos textos com dados diversos.

- Guia de Aprendizagem

Nesta aula vamos implementar os métodos de operação em Pilha Encadeada associados a manipulação de arquivos.

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos exemplos do GitHub da disciplina, responda a lista de exercícios. Lembre de colocar seus códigos no seu repositório da disciplina.

- Referência Bibliográfica

EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18).

3º aula / 3º semana

Modalidade da aula: Presencial - 4 de March de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os algoritmos de inserção, remoção, atualização e percurso de Listas Encadeadas: simples, duplas e circulares.

- Guia de Aprendizagem

Nesta aula vamos implementar os métodos de operação em Lista Encadeada simples

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos exemplos do GitHub da disciplina, responda a lista de exercícios. Lembre de colocar seus códigos no seu repositório da disciplina.

Veja os vídeos do professor sobre lista:

- #1 Estruturas de Dados Listas encadeadas: inserção com ordenação - [Clique aqui](https://youtu.be/3w8DWKQoqkQ)
- #2 Estruturas de Dados Listas encadeadas: 4 métodos básicos - <https://youtu.be/3w8DWKQoqkQ>
- #3 Estruturas de Dados Listas encadeadas: remoção - <https://youtu.be/UH3wkpjNgg>

- Referência Bibliográfica

EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18).

Vídeos do professor sobre lista:

#1 Estruturas de Dados Listas encadeadas: inserção com ordenação - [Clique aqui](#) #2 Estruturas de Dados Listas encadeadas: 4 métodos básicos - <https://youtu.be/3w8DWKQoqkQ>
#3 Estruturas de Dados Listas encadeadas: remoção - <https://youtu.be/UH3wkpjNgg>

4º aula / 4º semana

Modalidade da aula: Presencial - 11 de March de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os algoritmos de inserção, remoção, atualização e percurso de Listas Encadeadas: simples, duplas e circulares.

- Guia de Aprendizagem

Nesta aula vamos implementar os métodos de operação em Lista Encadeada dupla

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos de lista simples, construídos anteriormente, refatorá-los para listas duplas.

Veja vídeo do professor sobre listas:

#6 Estrutura de Dados: Listas Duplamente Encadeadas operações básicas - <https://youtu.be/J3bZFIBxznY>

- Referência Bibliográfica

EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18).

Vídeo do professor sobre listas:

#6 Estrutura de Dados: Listas Duplamente Encadeadas operações básicas - <https://youtu.be/J3bZFIBxznY>

5º aula / 5º semana

Modalidade da aula: Presencial - 18 de March de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os algoritmos de inserção, remoção, atualização e percurso de Listas Encadeadas: simples, duplas e circulares.

- Guia de Aprendizagem

Nesta aula vamos implementar os métodos de operação em Lista Encadeada circular

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos de lista simples ou dupla, construídos anteriormente, refatorá-los para listas circulares.

Ver os vídeos do professor sobre listas:

#4 Estruturas de Dados: Listas Encadeadas Circulares (percurso) - <https://youtu.be/Dyh3yvw2CKM>

#5 Estruturas de Dados: Listas Encadeadas Circulares (inserção) - <https://youtu.be/BdXmP4xKyto>

- Referência Bibliográfica

EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18).

Vídeos do professor sobre listas:

#4 Estruturas de Dados: Listas Encadeadas Circulares (percurso) - <https://youtu.be/Dyh3yvw2CKM>

#5 Estruturas de Dados: Listas Encadeadas Circulares (inserção) - <https://youtu.be/BdXmP4xKyto>

6º aula / 6º semana

Modalidade da aula: Presencial - 25 de March de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os algoritmos de inserção, remoção, atualização e percurso de Filas Encadeadas e DEQUES

- Guia de Aprendizagem

Nesta aula vamos implementar os métodos de operação em Fila

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos exemplos do GitHub da disciplina, responda a lista de exercícios. Lembre de colocar seus códigos no seu repositório da disciplina.

- Referência Bibliográfica

EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18).

7º aula / 7º semana

Modalidade da aula: Presencial - 1 de Abril de 2026

- Introdução

Fazer um programa que trate pontos em um plano cartesiano (x,y). Os pontos devem estar armazenados em um arquivo texto, um ponto abaixo do outro, em que x e y estão separados por ';'. O programa deve ler o arquivo e popular seus dados em uma lista encadeada dinamicamente. O usuário por sua vez, deve ser capaz de exibir esses dados e procurar por um ponto em especial.

- Repertório profissional

O programa deve ser construído no editor de codificação de preferência e ao final deve ser disponibilizado no GitHub pessoal do aluno.

- Produto de aprendizagem

Implementar um programa, em modo terminal, modularizado que manipule dados em arquivo texto e que sejam populados em lista encadeada dinamicamente.

8º aula / 8º semana

Modalidade da aula: Presencial - 8 de Abril de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os algoritmos de inserção, remoção, atualização e percurso de Filas Encadeadas e DEQUES

- Guia de Aprendizagem

Nesta aula vamos discutir e reavaliar o sistema construído no processo de avaliação.

- Encaminhamento próxima aula - Atividade Discente Efetiva

Corrija os erros e/ou equívocos existentes no código pessoal da avaliação.

- Referência Bibliográfica

EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18).

9º aula / 9º semana

Modalidade da aula: Presencial - 15 de Abril de 2026

Unidade 2 - Matrizes

- Experiência formativa

O que estudaremos?

Estudaremos a aplicação e a implementação de Matrizes de "N" dimensões, Matrizes especiais (diagonal, triangulares, simétrica e antissimétrica) e Matrizes esparsas

O que aprenderemos?

Aprenderemos como empregar as matrizes especiais em cenários adequados, como executar métodos que utilizem matrizes especiais em problemas e como construir programas computacionais que tratem de problemas que necessitem o uso de matrizes esparsas

Sobre o que refletiremos?

Como aplicar as matrizes especiais em problemas reais? Como implementar as matrizes esparsas nas linguagens C, C++, Java e C#? Como matrizes especiais são fundamentais em outras áreas da Computação?

- Competências

Manipular as diferentes matrizes especiais na construção de programas de computador para melhor otimização de memória.

- Objetivos

- 2.1 Empregar as matrizes especiais em cenários adequados;
- 2.2 Executar métodos que utilizem matrizes especiais em problemas;
- 2.3 Construir programas computacionais que tratem de problemas que necessitem o uso de matrizes esparsas.

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, matrizes N dimensões, especiais e esparsas.

- Guia de Aprendizagem

Nesta aula vamos discutir o que são matrizes de N dimensões, especiais e esparsas. Além disso, vamos aprender a declarar, criar, percorrer e preencher essas matrizes nas linguagens C e C++.

- Encaminhamento próxima aula - Atividade Discente Efetiva

Coloque todos os códigos produzidos do conteúdo no GitHub pessoal da disciplina.

- Referência Bibliográfica

GitHub da disciplina.

10º aula / 10º semana

Modalidade da aula: Presencial - 22 de April de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, matrizes N dimensões, especiais e esparsas.

- Guia de Aprendizagem

Nesta aula vamos implementar matrizes esparsas, operações de criação, inserção, busca e remoção, nas linguagens C e C++.

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos da disciplina disponibilizados no GitHub, faça a lista de exercícios.

- Referência Bibliográfica

GitHub da disciplina.

11º aula / 11º semana

Modalidade da aula: Presencial - 29 de April de 2026

Unidade 3 - Árvores

- Experiência formativa

O que estudaremos?

Estudaremos a implementação de estruturas de dados não lineares do tipo Árvore de maneira útil.

O que aprenderemos?

Aprenderemos compor soluções diversas que utilizem árvores como estruturas de dados para resolução de problemas, como projetar sistema que armazene dados em memória na estrutura árvore para problemas específicos na área da computação. Além disso, como combinar soluções computacionais que utilizem árvores encadeadas dinamicamente para resolução de problemas.

Sobre o que refletiremos?

Como aplicar as estruturas de dados não lineares (árvores) em problemas reais? Como implementar e/ou aplicar árvores nas linguagens C, C++, Java e C#? Como as estruturas de dados não lineares são fundamentais em outras áreas da Computação?

- Competências

Criar estruturas de dados não lineares do tipo árvores utilizadas em programação de maneira útil.

- Objetivos

- 3.1 Compor soluções diversas que utilizem árvores como estruturas de dados para resolução de problemas;
- 3.2 Projetar sistema que armazene dados em memória na estrutura árvore para problemas específicos na área da

computação;

3.3 Combinar soluções computacionais que utilizem árvores encadeadas dinamicamente para resolução de problemas.

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, métodos recursivos, com e sem retorno. Além disso, aprender como identificar no sistema operacional a pilha de processos recursivos.

- Guia de Aprendizagem

Nesta aula vamos discutir conceitos, aplicações e formas de métodos recursivos.

- Encaminhamento próxima aula - Atividade Discente Efetiva

Veja o vídeo sobre recursão do professor - https://youtu.be/5WAulVX_a0M

- Referência Bibliográfica

Vídeo do professor - https://youtu.be/5WAulVX_a0M

12º aula / 12º semana

Modalidade da aula: Presencial - 6 de May de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os algoritmos de inserção, remoção, atualização e percurso de Árvores Binárias de Pesquisa.

- Guia de Aprendizagem

Nesta aula vamos implementar os métodos de percurso em árvores.

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos exemplos do GitHub da disciplina, responda a lista de exercícios. Lembre de colocar seus códigos no seu repositório da disciplina.

- Referência Bibliográfica

EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18).

13º aula / 13º semana

Modalidade da aula: Presencial - 13 de May de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os algoritmos de inserção, remoção, atualização e percurso de Árvores Binárias de Pesquisa.

- Guia de Aprendizagem

Nesta aula vamos implementar o método de inserção em árvores binárias de pesquisa.

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos exemplos do GitHub da disciplina, responda a lista de exercícios. Lembre de colocar seus códigos no seu repositório da disciplina.

- Referência Bibliográfica

EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18).

14º aula / 14º semana

Modalidade da aula: Presencial - 20 de May de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os algoritmos de inserção, remoção, atualização e percurso de Árvores Binárias de Pesquisa.

- Guia de Aprendizagem

Nesta aula vamos implementar o método de remoção em árvores binárias de pesquisa.

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos exemplos do GitHub da disciplina, responda a lista de exercícios. Lembre de colocar seus códigos no seu repositório da disciplina.

Veja o vídeo do professor:

Estruturas de Dados : podar e excluir em árvores encadeadas - <https://youtu.be/HTjX4RgZc3w>

- Referência Bibliográfica

EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18).

Vídeo do professor:

Estruturas de Dados : podar e excluir em árvores encadeadas - <https://youtu.be/HTjX4RgZc3w>

15º aula / 15º semana

Modalidade da aula: Presencial - 27 de May de 2026

- Introdução

Fazer um programa que trate dados de glicemia de pacientes diabéticos (dia, hora, valor glicêmico). Os dados devem estar armazenados em um arquivo texto, um ponto abaixo do outro, em que dia, hora e valor estão separados por ';'. O programa deve ler o arquivo e popular seus dados em uma árvore binária de pesquisa, tendo como critério de ordenação a data. O usuário por sua vez, deve ser capaz de exibir esses dados e procurar por um valor glicêmico a partir do dia.

- Repertório profissional

O programa deve ser construído no editor de codificação de preferência e ao final deve ser disponibilizado no GitHub pessoal do aluno.

- Produto de aprendizagem

Implementar um programa, em modo terminal, modularizado que manipule dados em arquivo texto e que sejam populados em uma árvore binária de pesquisa.

16º aula / 16º semana

Modalidade da aula: Presencial - 3 de June de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os algoritmos de inserção, remoção, atualização e percurso de árvores

- Guia de Aprendizagem

Nesta aula vamos discutir e reavaliar o sistema construído no processo de avaliação.

- Encaminhamento próxima aula - Atividade Discente Efetiva

Corrija os erros e/ou equívocos existentes no código pessoal da avaliação.

- Referência Bibliográfica

EDELWEISS, Nina; GALANTE, Renata. Estruturas de dados. Porto Alegre, RS: Bookman, 2009. 261 p. (Série livros didáticos informática ufrgs ; 18).

17º aula / 17º semana

Modalidade da aula: Presencial - 10 de June de 2026

Unidade 4 - Grafos

- Experiência formativa

O que estudaremos?

Estudaremos a implementação de estruturas de dados não lineares do tipo Grafo de maneira útil.

O que aprenderemos?

Aprenderemos compor soluções diversas que utilizem grafos como estruturas de dados para resolução de problemas, como projetar mapas rodoviários e seus percursos, contextualizando e aplicando a problemas específicos na área da computação.

Sobre o que refletiremos?

Como aplicar as estruturas de dados não lineares (grafos) em problemas reais? Como implementar e/ou aplicar grafos nas linguagens C, C++, Java e C#? Como as estruturas de dados não lineares são fundamentais em outras áreas da Computação?

- Competências

Gerar estruturas de dados não lineares do tipo grafo utilizadas em programação de maneira útil .

- Objetivos

- 4.1 Compor soluções diversas que utilizem grafos como estruturas de dados para resolução de problemas na área da computação;
- 4.2 Projetar sistema que faça roteamento ou percurso via uso de grafos para problemas específicos na área da computação;
- 4.3 Combinar soluções computacionais que utilizem grafos para mapeamento e roteamento.

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, grafos alocados dinamicamente em memória. Além disso, os métodos de percurso Profundidade e Amplitude.

- Guia de Aprendizagem

Nesta aula vamos discutir conceitos, aplicações de grafos e seus métodos de percurso.

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos exemplos no GitHub da disciplina, baixe, execute e analise os códigos.

- Referência Bibliográfica

GitHub da disciplina

18º aula / 18º semana

Modalidade da aula: Presencial - 17 de June de 2026

- Pontos de Aprendizagem

Identificar, declarar e implementar, nas linguagens C e C++, os métodos de percurso Profundidade e Amplitude e associa-los a forma de armazenamento por matriz de adjacência.

- Guia de Aprendizagem

Nesta aula vamos implementar grafos que representem mapas rodoviários, a fim de aplicar diferentes percursos por força bruta (profundidade e amplitude).

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos exemplos no GitHub da disciplina, baixe, execute e analise os códigos.

- Referência Bibliográfica

GitHub da disciplina

19º aula / 19º semana

Modalidade da aula: Presencial - 24 de June de 2026

- Introdução

Fazer um programa, em modo terminal, que represente um mapa qualquer entre cidades, por exemplo, e que o usuário possa fazer diferentes percursos via os métodos amplitude e profundidade.

- Repertório profissional

O programa deve ser construído no editor de codificação de preferência e ao final deve ser disponibilizado no GitHub pessoal do aluno.

- Produto de aprendizagem

Implementar um programa, em modo terminal, modularizado que manipule um mapa de cidades, em que o usuário possa cadastrar cidades e estradas, e em seguida realizar consultas de rotas entre cidades.

20º aula / 20º semana

Modalidade da aula: Presencial - 1 de July de 2026

- Pontos de Aprendizagem

Implementar, nas linguagens C e C++, árvore geradora mínima e o método de balanceamento em árvores binárias de

pesquisa.

- Guia de Aprendizagem

Nesta aula vamos implementar árvore geradora mínima e o método de balanceamento de árvores binárias de pesquisa.

- Encaminhamento próxima aula - Atividade Discente Efetiva

A partir dos códigos exemplos no GitHub da disciplina, baixe, execute e analise os códigos.

- Referência Bibliográfica

GitHub da disciplina